Improving Spherical-Attribute-Image Method for Complicated Object with Depressions and Partial Fractures

窪み箇所や部分欠損を含む複雑な形状の物体のための SAI 法の改良

by

Naoya Saga

佐賀 直也

A Master Thesis

修士論文

Submitted to
The Department of Computer Science
The Graduate School of Information Science and Technology
The University of Tokyo
February 2006
In partial fulfillment of the requirements
For the Degree of Master of Information Science and Technology

Thesis Supervisor: Katsushi Ikeuchi **池内 克史**

Abstract

In 3-dimensional image analysis, the methods through converting shape information of the target object into the unified data format, which is completely independent of its shape, are widely used. The Spherical-Attribute-Image (SAI) method is one of such methods. In the SAI method, a 3-dimensional shape model is first approximated with the mesh structure. Next geometrical attribute defined on each point of the mesh is mapped onto a spherical surface. Shape information of any model is expressed on the spherical image. The image, called *SAI*, is used for the analysis.

In the process of constructing the SAI, the deformable surface method is used. The method deforms the spherical mesh into the model by converging some metric about difference between the two. In the case that the model is quite different from a sphere, for example the model including some depressions, however, the wrong convergence happens and resultantly prevents to obtain an appropriate SAI. Then, we propose a method for leading to the correct convergence using the intermediate shape model between the model and a sphere.

In the SAI method, the SAI obtained from some model may be quite different from the image obtained from the partially lacking model, especially in the case lacking regions are bigger. That has an adverse affect on the analysis. Then, we propose a method for reducing the affect by appropriately weighting the difference of the geometrical attribute between two SAIs.

In actuality, we measured skulls of fowls with a laser scanner and generated precise 3-dimensional shape models of their skulls. Using these models we generated and analyzed their SAIs, and estimated phylogenetic relationships between related breeds of the fowls for evaluating validity of our proposed method.

論文要旨

３次元画像解析において，解析対象物体の特徴を物体形状に全く依存しない統一されたデータ形式に変換し，解析を行う手法が広く用いられている．その手法の一つである Spherical Attribute Image (SAI) 法では，３次元形状モデルをメッシュ構造で表面近似し，メッシュ上の各点で定義される幾何学的特徴量を球面上にマッピングし，形状情報を球面上で表現した SAI と呼ばれるデータ形式に変換し，解析を行う．

データ変換の過程において，球状メッシュを変形しモデル表面に収束させていく Deformable Surface の手法が用いられているが，窪み箇所を含むモデルのような，球と形状が大きく異なるモデルでは誤収束のため好ましい変換を得ることが困難である．そこで，球とモデルの中間的な形状データを用いて誤収束を回避する手法を提案する．

また SAI 法では，ある形状モデルから得られる SAI と，そこから一部分が欠損したモデルから得られる SAI とでは，欠損の程度によっては大きく異なってしまい，解析に悪影響を与えるという問題がある．そこで，SAI 間の幾何学的特徴量の差に適切な重み付けを行うことにより，欠損の影響を軽減する手法を提案する．

実際に，レーザースキャナを用いて鶏の頭蓋骨標本を測定し生成した３次元形状モデルを用いて，提案手法により SAI を生成，解析し，鶏の種間の系統関係の推定を行い，本手法の有用性を示す．

Acknowledgements

I extend my sincere appreciation to my thesis supervisor, Professor Katsushi Ikeuchi, who introduced me to the various computer vision techniques essential for designing my analysis methods.

In addition, I would like to thank members of the Computer Vision Laboratory, Institute of Industrial Science, the University of Tokyo; in particular, I would like to thank Dr. Jun Takamatsu, and Ms. Uehara, for their advice, help, and encouragement.

Last, but not least, I would like to thank my family for their support during my time as a student.

Contents

Chapter 1 Introduction

About the shape information of a 3-dimensional shape model which is measured in any way, 3-dimensional analysis is a method that we analyze its structure, let it lack unnecessary information and extract characteristic information. About the main purposes of doing 3-dimensional analysis, we list the industrial purpose, the medical purpose, and so on. We give examples which are quality control by checking the shape of products and the transformation of the examples, and we support for medical practice by measuring and analyzing a part of the human body shape. 3-dimensional analysis is essential for our life now.

The spherical attribute image (SAI) method is a method for analyzing 3-dimensional shape by mapping a geometrical attributes of 3-dimensional mesh model onto a sphere in order to represent the entire shape of a 3-dimensional object. The attributes are defined on each point of the mesh and the SAIs are invariant against translation, rotation and scaling of the object. With the SAI, by using the "simplex angles" as the geometrical attributes mapping onto a sphere, it is possible to reconstruct the original 3-dimensional shape from the SAI only. Moreover, since the SAIs are invariant against translation, rotation and scaling, it is possible to compare the difference of object shapes by comparison of the SAIs whether the original object had been translated and rotated. By linear interpolation of the attributes on the different SAIs, the synthetic 3-dimensional shape can be reconstructed. This synthetic shape is the neutral shape of the multiple origins.

To generate SAIs, some methods of mapping geometrical attributes defined at each point of 3-dimensional models onto spheres have been proposed, including: Gauss Mapping [1], Extended Gaussian Image (EGI) [2] and Complex Extended Gaussian Image (CEGI) [3], etc.

These method can only treat the objects which are convex and topologically equal to a sphere. However, objects which are usually analyzed are not satisfied with this condition. Because of this, we use SAI method in this thesis.

For generating SAIs, we first need to approximate surface of 3-dimensional model by

semi-regular mesh model. Semi-regular mesh is the mesh satisfied with the following two features. One is homogeneity, that is, all nodes of the mesh have three neighbor nodes. The other is local regularity, that is, all distances between any two neighbor nodes are equal. In this thesis, we used the Deformable Surface Method. This method deforms a spherical original mesh to fit to the object's surface by iterative calculation. With regard to the deformable surface method, some related research was carried out: Irregular meshes [4], finite element models [4], balloon models [5] and the applications of medical imaging [6].

However, in practice, if the mesh converges onto a part of depression by the original method [7], some unfavorable cases are happened. Because the mesh covers over the depression and does not fit onto the area, the mesh is satisfied with local regularity and bad approximation of surface of 3-dimensional model. Otherwise, because the mesh approximates the depression of surface of 3-dimensional model by emphasizing the parameter of the mesh approximation, the mesh has good approximation of surface of 3-dimensional model and no local regularity.

The cause of the problem is to realize the two features by one step. It is difficult to obtain the mesh with two features: good surface approximation of 3-dimensional shape model and local regularity. We propose this new method, and examine this advantage. And, for solving this problem, we obtain one feature by one step. After all, we realize two feature by two steps. Concretely speaking, (1) we try to obtain good surface approximation of 3-dimensional shape model, and generate a series of models whose shapes are gradually changing from initial sphere mesh to the boundary of 3-dimensional shape model, and (2) we fit the sphere semi-regular mesh onto a series of models in order considering the local regularity. Moreover, these models from initial sphere mesh to the boundary of 3-dimensional shape model are named "guiding intermediate shapes".

Moreover, using SAIs generated from 3-dimensional shape models, we are able to do 3-dimensional analysis among any 3-dimensional models. Because the geometrical attributes are mapped onto SAI's surface, we define the dissimilarity between two SAIs by comparing their geometrical attributes. And, we define the distance between

two 3-dimensional shape models by taking advantage of the dissimilarity between two SAIs. Moreover, assuming the distances among several 3-dimensional shape models to be similarities, and comparing similarities, we can classify several 3-dimensional shape models by statistical classifier, e.g. hierarchical cluster analysis [13]. Finally, we generate dendrogram.

The thesis is organized as follows: In Chapter 2, we describe the concept and algorithm of the spherical attribute image (SAI) method that is one of the methods for mapping geometrical attributes of 3-dimensional shape models onto spheres. In Chapter 3, we first pick up the problem of the former method of generating SAIs. Next, we propose the new method and prove the advantage of our proposed method. In Chapter 4, we define the distance between two 3-dimensional shape models by taking advantage of two SAIs. Moreover, we analyze several 3-dimensional shape models, and generate dendrogram. Finally, in Chapter 5, we conclude this thesis and describe future works.

Chapter 2 Spherical Attribute Image (SAI)

In this chapter, we describe the concept of the spherical attribute image (SAI) method, one of the methods for mapping the geometrical attributes of the 3-dimensional mesh surface onto a sphere; we then describe the algorithm for generating an SAI.

2.1 The concept of the Spherical Attribute Image (SAI) method

The spherical attribute image (SAI) method is a method for mapping a geometrical attributes of 3-dimensional mesh model onto a sphere in order to represent the entire shape of a 3-dimensional object. The attributes are defined on each point of the mesh and the SAIs are invariant against translation, rotation and scaling of the object.

With the SAI, by using the "simplex angles" as the geometrical attributes mapping onto a sphere, it is possible to reconstruct the original 3-dimensional shape from the SAI only.

Moreover, since the SAIs are invariant against translation, rotation and scaling, it is possible to compare the difference of object shapes by comparison of the SAIs whether the original object had been rotated.

By linear interpolation of the attributes on the different SAIs, the synthetic 3-dimensional shape can be reconstructed. This synthetic shape is the neutral shape of the multiple origins.

2.1.1 Representation of a 2-dimensional curve

Before representation of a 3-dimensional shape, it is helpful to understand how one can represent a 2-dimensional shape. Now, there is a sequence of 2-dimensional points as the object representation. A natural way of representing this shape is the approximation of their boundaries to the list of line segments as shown in Fig. 2.1.

Here, the line segments have equal lengths. The connected points of the lines are called "nodes;" one can discretely map the geometrical attributes of all nodes onto a circle while keeping the sequence of the nodes (See Fig. 2.2).

Since they are mapped onto a circle, these attributes are invariant against rotation

and scaling. If the line segments have sufficient density, the representation can make the same attribute-mapped circle as the object that had been rotated. Scaling is the same.

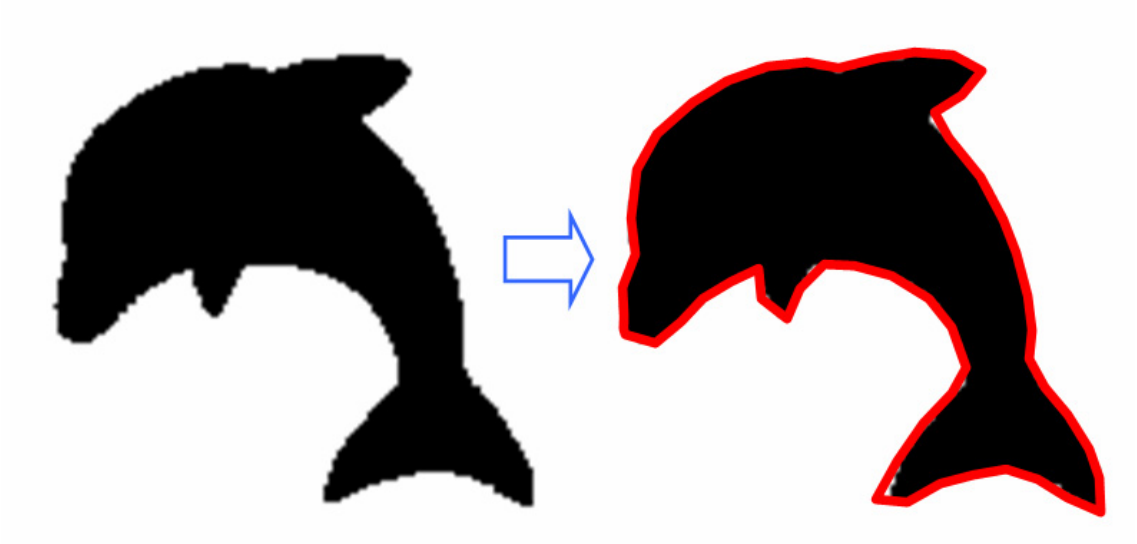


Figure 2. 1: Approximation of 2-dimensional shape by the line segments

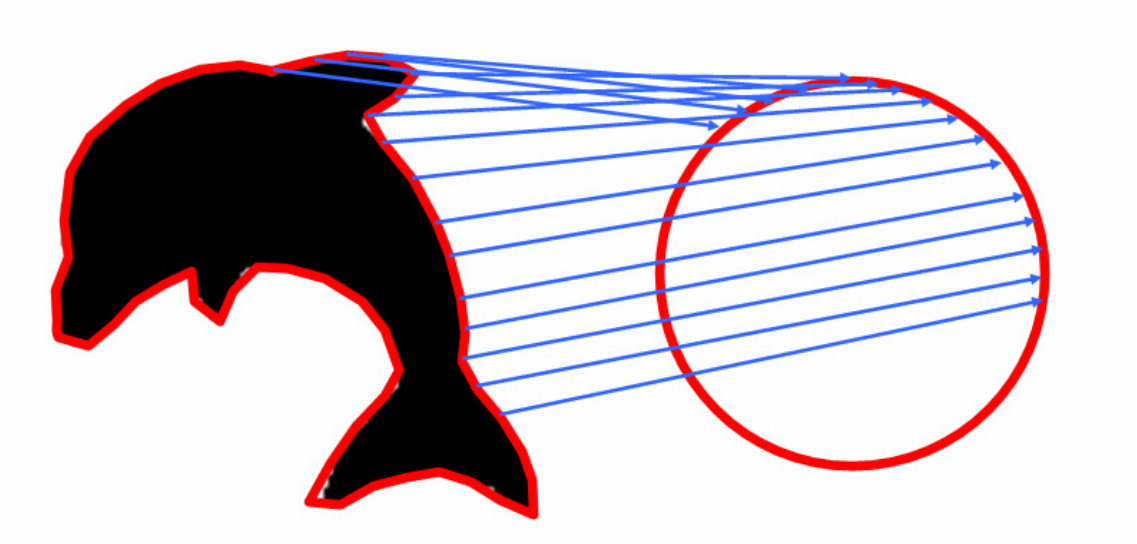


Figure 2. 2: Mapping the Attribute onto a circle

In this representation, each geometrical attribute is the "turning angle" that is one of the discrete curvatures. The turning angles are defined at each node of the line segments, and they are exterior angles of the two lines connected to the node (See Fig.

2.3).

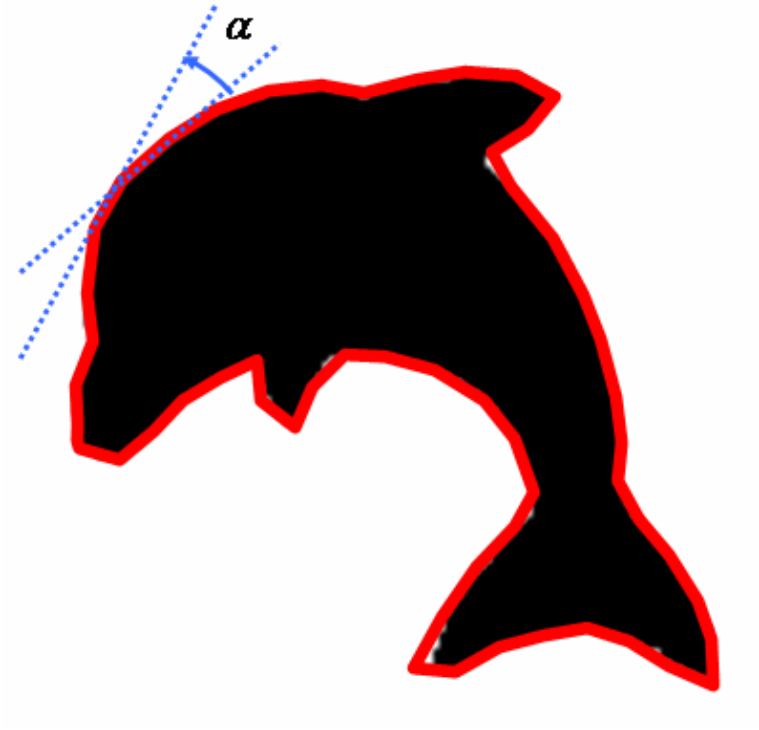


Figure 2. 3: Definition of turning angle

Under this definition, a turning angle α is positive when the curve at the node is convex; α is negative when the curve is concave, and α is zero when the curve is flat.

In addition, since these attributes are defined by only the relative position of the node and the two neighbor nodes, they are invariant against rotation. And since these attributes are angles, they are invariant against scaling. These attributes are suitable for this method of shape representation.

The algorithm of this method of comparing multiple 2-dimensional shapes is shown below:

(1) Approximate a boundary of a 2-dimensional shape to a list of line segments

(2) Calculate the turning angles as geometrical attributes at all nodes

(3) Map the attributes onto a circle, keeping their sequence

(4) Turn the circles to minimize the difference of the attributes that are mapped at the same place on the circle

(5) Compare the attributes at each place on the circle

Then, the circle on which the turning angles are mapped can reconstruct the original 2-dimensional shape. If the positions of two neighbor nodes P_1, P_2 are known, a node P has to exist on the perpendicular bisector of the line connected between the two neighbors. And, from the definition of the turning angle, the absolute value of the turning angle is proportional to the distance between P and the line connecting the two neighbors. That is, the position of a node P can be determined by the turning angle at the node P.

From these facts, if it has the same number of nodes, arbitrary line segments can transform to the original shape by minimizing the "error" iteratively at all nodes simultaneously; here, the error is the distance between a position of the node and the "true" position of the node that is determined by the positions of the two neighbor nodes and the turning angle.

In this method, generation of the attribute-mapped circle from the shape is called "forward mapping," and reconstruction of the original shape from the attribute-mapped circle is called "inverse mapping."

2.1.2 Expansion to 3-dimensional surfaces

The SAI method is the direct expansion to 3-dimension of the method of representing 2-dimensional shapes described in the previous section. The "SAI" is a sphere on whose surface the geometrical attributes of the original 3-dimensional shapes are mapped.

In the SAI method, first, the "semi-regular" mesh model approximates the boundary of 3-dimensional shape. The semi-regular mesh is the mesh that is not self-interacting, and all the vertices have exactly three neighbor vertices. The mesh is a direct expansion of the line segments; all their nodes have exactly two neighbor nodes.

In this method, the mesh has to satisfy a constraint called "local regularity." This constraint is the expansion to 3-dimension of the constraint that all line segments have equal length. We will describe local regularity in a 3-dimension case in Section 2.2.3.

Next, the "simplex angle," one of the geometrical attributes, is calculated at each node

of the mesh. The attribute is an expansion to 3-dimension of the turning angle. Like the turning angle, the simplex angle can be calculated only by relative positions of the node and its three neighbors; its sign shows whether the surface is convex or concave, and its absolute value is proportional to the distance between the node and the triangle that is made by the three neighbors. We will describe the simplex angle in Section 2.2.1.

In the SAI method, as in the case of 2-dimension, "forward mapping" is the generation of the attribute-mapped sphere from the 3-dimensional shape, and "inverse mapping" is the reconstruction of the original 3-dimensional shape from the attribute-mapped sphere.

2.2 The SAI Algorithm

In this section, we describe the details of the SAI algorithm. We first describe the definition of the simplex angle. Next, we describe how to make the geodesic dome be the initial spherical mesh for deforming, and how to deform the mesh to the object's surface. Then, we describe how to map the simplex angle calculated at each mesh node on the sphere using the property of deformable surface that the mesh nodes before and after deforming have one-to-one correspondences.

2.2.1 Simplex Angle

The simplex angle is a geometrical attribute that calculated at each node of a semi-regular mesh by the relative positions of the node and its three neighbor nodes.

Let P be the position of a node, and P_1, P_2, P_3 be the positions of three neighbor nodes of P. And let O be the center of the circumscribed sphere of the tetrahedron $P\,P_1\,P_2\,P_3$, C be the center of the circumcircle of the triangle $P_1\,P_2\,P_3$. And let Z be a line connected O and C, and Π be a plane that includes Z and P (See Fig. 2.4). Moreover, consider a section plane of the circumscribed sphere of the tetrahedron $P\,P_1\,P_2\,P_3$ by the plane Π.

On the section plane, the circumcircle of the triangle $P_1\,P_2\,P_3$ becomes the subtense of P. The exterior angle ϕ of P on the plane is the simplex angle at the mesh node P

(See Fig. 2.5).

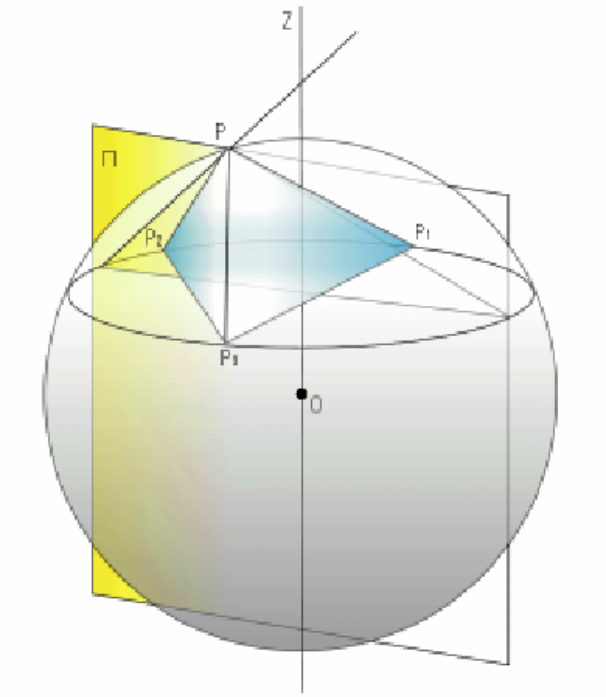


Figure 2. 4: The concept of simplex angle

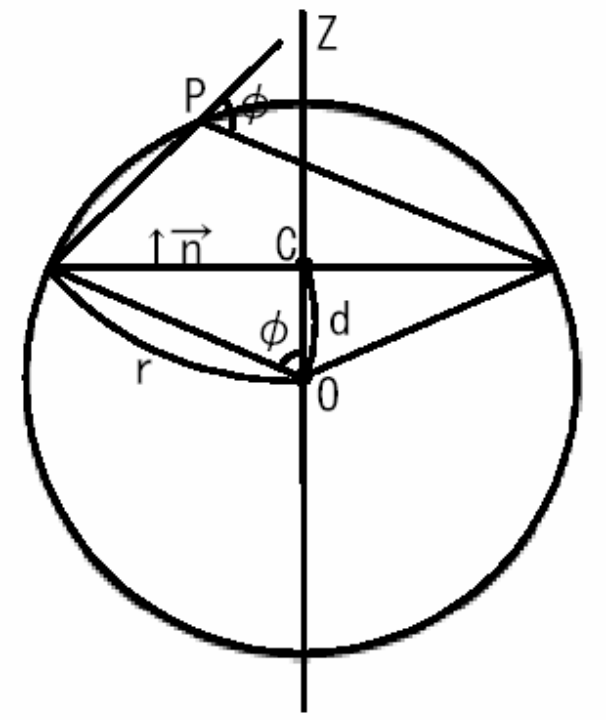


Figure 2. 5: Definition of simplex angle

From these definitions, the simplex angle ϕ can be written as Equation (2.1), where r is the radius of the circumcircle of the triangle $P_1\ P_2\ P_3$, R is the radius of the circumscribed sphere of the tetrahedron $P\ P_1\ P_2\ P_3$, and $\vec{N}$ is the normal vector of the triangle $P_1\ P_2\ P_3$.

$$\begin{aligned} \cos\phi &= \frac{\left|\overrightarrow{OC}\right|}{R} sign\left(\overrightarrow{OC}\cdot\vec{N}\right) \\ \sin\phi &= \frac{r}{R} sign\left(\overrightarrow{PP_1}\cdot\vec{N}\right) \end{aligned} \tag{2.1}$$

Note that the domain of ϕ is $[-\pi,\pi]$.

Under this definition of the simplex angle, like the turning angle, because this geometrical attribute is an angle that is calculated only by the relative positions of the node and its neighbors, the simplex angle is invariant against rotation and scaling.

Moreover, when the simplex angle is positive, the node P is over the neighbor triangle $P_1\ P_2\ P_3$. When it is negative, P is under the triangle. And when it is zero, P is on the triangle. That is, when the simplex angle is positive/ negative/ zero, the surface at the node is convex/ concave/ flat. And, from the definition, the absolute value of the simplex angle is proportional to the distance between the node P and the neighbor triangle $P_1\ P_2\ P_3$.

These features of the simplex angle show that this angle is a natural expansion to 3-dimension of the turning angle. The SAI is generated by calculating this simplex angle at each node of the shape-approximated mesh and by mapping them onto the sphere.

2.2.2 Geodesic Dome: generating the initial mesh

The initial mesh of the deformable surface has to be a sphere-circumscribed semi-regular mesh. In this thesis, we first generated a sphere-circumscribed regular mesh named the “geodesic dome,” and dualized it onto the semi-regular mesh.

Generally, geodesic domes are sphere-circumscribed polyhedrons made of triangles. There are thousands of such polyhedrons, but for generating a more symmetrical mesh, it is desirable that: (1) the ratio of the lengths of three sides of the triangles is nearly

equal to 1, that is, each triangle of the polyhedron is nearly equal to a regular triangle (2) the triangles are as similar in shape as possible. In the sphere-circumscribed polyhedrons, it is well-known that there are three polyhedrons that completely fulfill the conditions, that is, there are three regular solids as shown in Fig. 2.6 made of only regular triangles: the regular tetrahedron, the regular octahedron, and the regular dodecahedron.

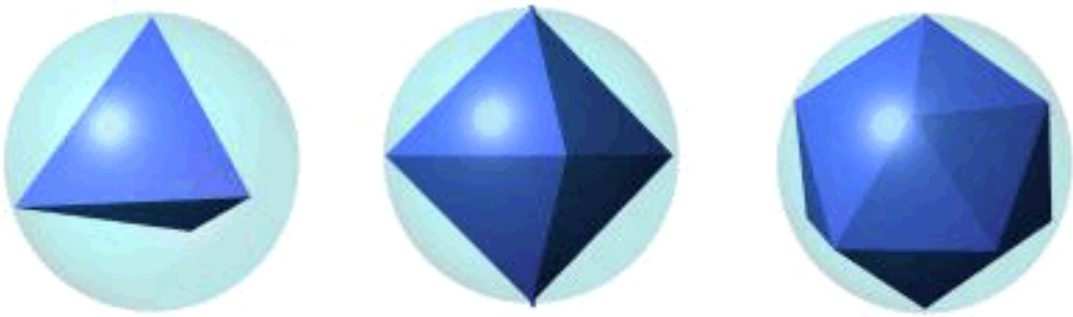


Figure 2. 6: Regular tetrahedron, octahedron and dodecahedron

Since, in these three polyhedrons, the regular dodecahedron is most similar to the sphere, dividing its facial triangles can generate a more sphere-similar geodesic dome. First, bisecting three edges of each triangle and generating two new lines connecting between the new point and the original points makes the polyhedron more sphere-similar. Therefore each original triangle is divided into four triangles. Because this new polyhedron is not circumscribed to the sphere, the newly generated points (the medians of three edges) have to project from the center of the sphere onto the surface of the sphere. As a result, this operation divides each triangle into one regular triangle and three isosceles triangles. By repeating this operation at all vertices of the polyhedron, a more sphere-similar geodesic dome can be generated. Figure 2.7 shows the aforementioned process. In the figure, a red triangle is a regular triangle.

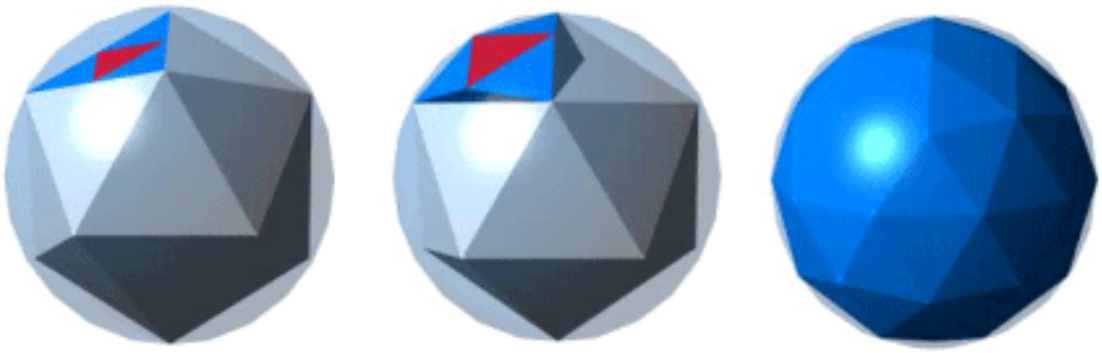


Figure 2. 7: The processes of dividing the geodesic dome

Iterating this operation can generate the geodesic dome with arbitrary density. After iterating n-times, the generated geodesic dome has 20×4^n triangles. In this thesis, we used the geodesic dome with $n = 5$ shown in Fig. 2.8 as the initial mesh for generating SAIs. This geodesic dome is a polyhedron that contains 20480 triangles.

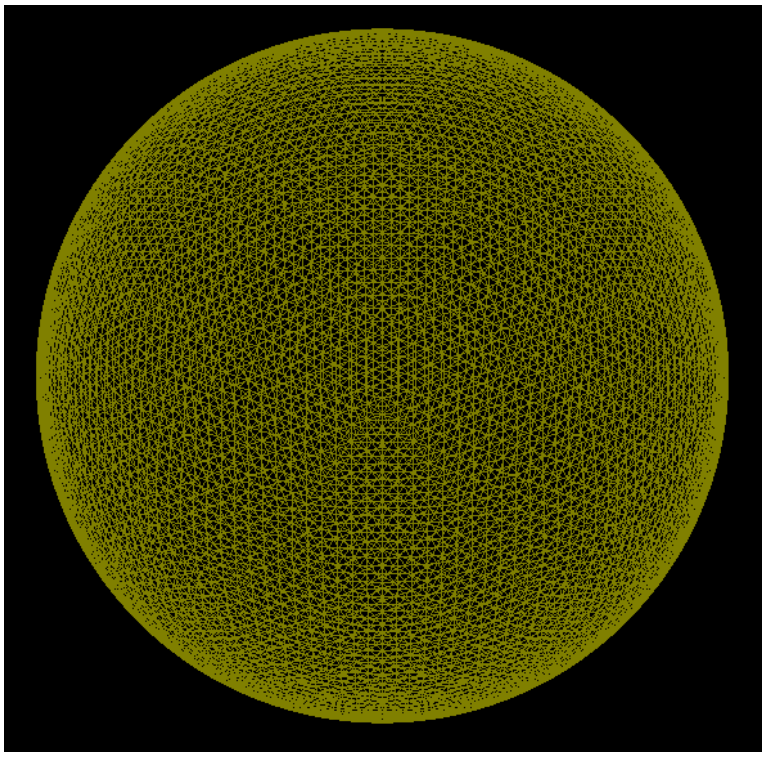


Figure 2. 8: Geodesic dome that contains 20480 triangles

Next, calculating the dual mesh of the geodesic dome (notice that it is a regular mesh) can generate a semi-regular and sphere-similar mesh. In this case, the generated

semi-regular mesh has 20480 vertices; all the mesh nodes have exactly three neighbor nodes, and this mesh is made up of many hexagons and very few pentagons (See Fig. 2.9).

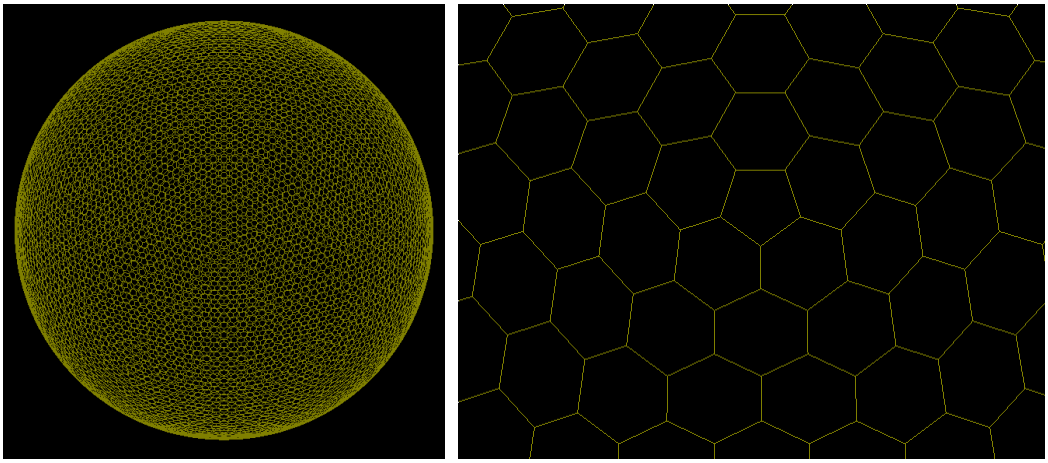


Figure 2. 9: Generated dual mesh and its macrograph

2.2.3 Deformable Surface

In this section, we describe how to deform the initial mesh that is the spherical semi-regular mesh described in the previous section to the 3-dimensional objective model. The term "model" means the object whose shape is approximated. This method assumes that the barycenter of the initial mesh is equal to the barycenter of the objective model and that the initial mesh is large enough to fully cover the model.

Now, consider the two imaginary "forces;" they can be calculated by the relative positions between the model and each node of the mesh, and by the "smoothness" of the mesh itself. And these forces independently move each node. By iteratively solving the equation of motion about these forces, the mesh is deformed to the model.

In this method, it is important that the mesh's geometrical structure is retained during the deformation, that is, that the three neighbors of each node are fixed. Therefore, by using a deformable surface, each node of the initial mesh and each node of the deformed mesh have one-to-one correspondences. These correspondences make it possible that the attributes defined at each node of the deformed mesh are mapped onto the initial spherical mesh.

Moreover, by keeping a constraint named “local regularity” during deformation, the original shape can be reconstructed from the SAI that is an attribute-mapped sphere. These features are products of the force defined by the smoothness of the mesh itself, that is, the force is the “internal force” described in the following section.

Local regularity is the expansion to 3-dimension of the constraint that all line segments have equal lengths. Specifically, at each node of the mesh, this constraint is that the foot perpendicular from the node to the neighbor triangle is equal to the barycenter of the neighbor triangle. That is, as shown in Fig. 2.10, let P be the node of the mesh, and P_1, P_2, P_3 be the three neighbor nodes. And consider the tetrahedron $P\ P_1\ P_2\ P_3$. Local regularity is the constraint that the foot Q of perpendicular from P to triangle $P_1\ P_2\ P_3$ is equal to the barycenter G of triangle $P_1\ P_2\ P_3$.

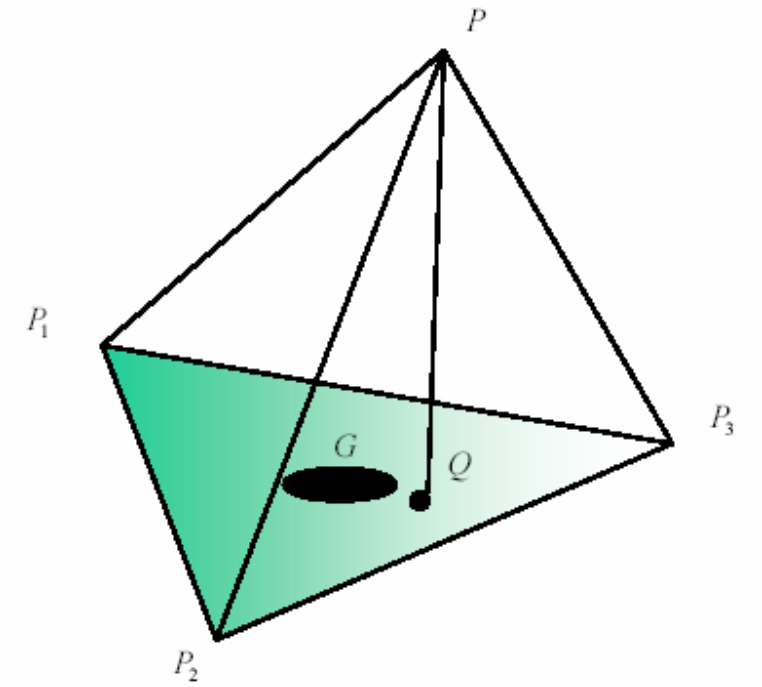


Figure 2. 10: Definition of local regularity

The algorithm of deformable surface can be written as follows:

(1) Calculate the “forces” at each node of the mesh

(2) Move each node by the equation of motion about the forces

(3) If the sum of distances between each node and the model is smaller than the threshold, this algorithm is finished

(4) Otherwise, go back to Step (1). Step (1) and (2) are repeated until the sum of distances between each node and the model is smaller than the threshold.

The equation of motion at the ith node P_i of the mesh can be written as Equation (2.2).

$$\frac{d^2 P_i}{dt^2} + k\frac{dP_i}{dt} = \alpha\overrightarrow{F_{Ext}} + \beta\overrightarrow{F_{Int}} \tag{2.2}$$

Here, $\overrightarrow{F_{Ext}}$ is the force named "external force" that is defined by the distance between the node and the model, and this force makes the node closer to the model and deforms the shape of the mesh. $\overrightarrow{F_{Int}}$ is the force named "internal force" that is defined by the internal structure of the mesh itself, and this force makes the mesh keep the structure of the semi-regular mesh, as well as local regularity.

α, β are the coefficients for adjusting the influences of two forces. If α is too large, the shape of the mesh becomes more similar to the model but the mesh cannot keep the structure of a semi-regular mesh and local regularity. On the other hand, if β is too large, the shape of the mesh becomes the shape that completely retains the structure and the constraint, that is, the shape becomes a sphere. There is a trade-off between accuracy of shape approximation and regularity of the mesh. In the following section, we concretely describe the external and internal forces.

(1) External Force (Data Force)

The external force is defined by the relative positions of the node and the model, and this force deforms the mesh to the shape of the model. This force considers the model to be a simple set of the 3-dimensional points, and does not use its connectivity.

Consider the ith node P_i of the mesh. If the "closest" point $M_{cl(i)}$ of P_i in the model is given, the external force $\overrightarrow{F_{Ext}}$ at P_i is defined as Equation (2.3), where $\overrightarrow{N_i}$ is the normal vector of the neighbor triangle of the node P_i.

$$\overrightarrow{F_{Ext}} = G\left(\frac{\left|\overrightarrow{P_i M_{cl(i)}}\right|}{D}\right)\left(\overrightarrow{P_i M_{cl(i)}} \cdot \overrightarrow{N_i}\right)\overrightarrow{N_i} \tag{2.3}$$

Here, the term “closest” means that the Euclidean distance between the two points is closest. $G(x)$ is 1 when the parameter x is 1 or less, and decrease rapidly when x is over 1. D is the threshold for judging the correspondence between the node and the closest point in the model. If the distance between the two points is larger than D, the correspondence may be incorrect; this value has to be adjusted according to the scale. The suitable choice of the $G(x)$ and D reduces the situation that the deforming falls into local minima. In this thesis, we used the function shown in Equation (2.4) as $G(x)$.

$$G(x)=\begin{cases}1 & (0 \le x \le 1)\\ \dfrac{1}{x^2} & (x>1)\end{cases} \tag{2.4}$$

Also, projecting the external force on the direction of $\overrightarrow{N_i}$ reduces the negative effect of the external force on the local regularity.

Generally, if the model is the very dense model such as the one scanned with a high-performance laser scanner, it is difficult to search the closest point $M_{cl(i)}$ in the model. Let N be total number of the points in the model and let n be number of the nodes of the mesh, the naive all-search algorithm costs $O(N)$ at each node in computational time.

But, in the deformable surface method, only the mesh is changed and the model is fixed during deformation. Therefore, by using k-d tree data structure for keeping the positions of the points, the computational cost of searching can be reduced. Accordingly, calculation of external force at each node costs $O(\log N)$ in computational time, the total computational cost at each stage of the iteration is $O(n \log N)$

(2) Internal Force (Smoothness Force)

The internal force is defined by the relative positions between the node and the three neighbor nodes; the force makes the mesh keep its local regularity. In Section 2.1, we describe how the circle, the point on which is mapped turning angles, can reconstruct the original shape.

We have described that the SAI method is the expansion to 3-dimension of this method,

and that local regularity is the expansion of the constraint that all line segments have equal lengths. In the case of 2-dimension, it is because the node has to be bound on a specific line that the constraint is required.

In an SAI, let P be a node of the mesh, let P_1, P_2, P_3 be the three neighbors of P, and Q be the foot of perpendicular from P to the triangle $P_1\ P_2\ P_3$. The simplex angle mapped on P can determine the distance between P and the triangle $P_1\ P_2\ P_3$. That is, if the simplex angle and the position of Q are given, the 3-dimensional position of P can be calculated, because P is bound on the perpendicular of $P_1\ P_2\ P_3$ that passes Q.

Now, since Q is a point on the triangle $P_1\ P_2\ P_3$, the position of Q can be represented as follows.

$$\begin{aligned} Q &= \varepsilon_1 P_1 + \varepsilon_2 P_2 + \varepsilon_3 P_3 \\ \varepsilon_1 &+ \varepsilon_2 + \varepsilon_3 = 1 \end{aligned} \tag{2.5}$$

$\varepsilon_1, \varepsilon_2, \varepsilon_3$ are called "metric parameters." The position of P can be determined by the positions of its three neighbors P_1, P_2, P_3, the simplex angle, and metric parameters. As the result, "local regularity" brings Q close to the neighbor triangle's barycenter $G = \frac{1}{3}P_1 + \frac{1}{3}P_2 + \frac{1}{3}P_3$, that is, it brings metric parameters close to $\varepsilon_1 = \varepsilon_2 = \varepsilon_3 = \frac{1}{3}$.

Finally, the internal force $\overrightarrow{F_{Int}}$ at the node P can be defined as Equation (2.6).

$$\overrightarrow{F_{Int}} = (G - Q) \tag{2.6}$$

2.2.4 From Shape to Attribute: Forward Mapping

In this section, we concretely describe the method to calculate the equations of motion. From the discussions in the previous sections, the equation of motion of the ith node P_i of the mesh can be written as Equation (2.7).

$$\frac{d^2 P_i}{dt^2} + k\frac{dP_i}{dt} = \alpha\overrightarrow{F_{Ext}} + \beta\overrightarrow{F_{Int}} \tag{2.7}$$

Since this equation is a continuous differential equation, it is difficult to solve it analytically. Therefore, we solved it numerically by using the Euler method that is one of the discretized numerical methods. The Eular method solves the equation by

iteratively calculating Equation (2.8), where $P_i^{(t)}$, $\overrightarrow{F_{Ext,i}^{(t)}}$, and $\overrightarrow{F_{Int,i}^{(t)}}$ are the ith node, its external force, and its internal force at the ith iteration, respectively.

$$P_i^{(t)} = P_i^{(t-1)} + (1-k)\left(P_i^{(t-1)} - P_i^{(t-2)}\right) + \overrightarrow{F_{Ext,i}^{(t)}} + \overrightarrow{F_{Int,i}^{(t)}} \tag{2.8}$$

By iteration of this calculation until the error function E is smaller than the threshold, this method can generate a mesh that is fitted to the model. E is the sum of all distances between each node P_i and its closest point $M_{cl(i)}$, that is, can be written as Equation (2.9).

$$E = \sum_i \left| \overrightarrow{P_i M_{cl(i)}} \right| \tag{2.9}$$

2.2.5 From Attribute to Shape: Inverse Mapping

In this section, we describe how we reconstructed the original shape from an SAI. In forward mapping, the two forces affect each node of the mesh: external force and internal force. The internal force is similar to that one in forward mapping. Concretely, it can be written as $\overrightarrow{F_{Int}} = (G - Q)$.

But, in the inverse mapping, the external force is the force to keep the distance between the node P and its neighbor triangle $P_1\ P_2\ P_3$ at the distance calculated by the simplex angle. As we have described in section 2.2.1, the simplex angle is proportional to the distance between the node P and the triangle $P_1\ P_2\ P_3$.

Here, the distance t can be calculated by solving Equation (2.10) which is a transformation of the equation of simplex angle, where r is the radius of the circumcircle of the triangle $P_1\ P_2\ P_3$, and l is the distance between the center of the circumscribed sphere of the tetrahedron $P\ P_1\ P_2\ P_3$ and the foot of the perpendicular from P to the neighbor triangle $P_1\ P_2\ P_3$.

$$\phi = \arctan\frac{t}{r-l} + \arctan\frac{t}{r+l} \tag{2.10}$$

Therefore, the distance t can be written as Equation (2.11), where ϕ is the simplex angle at the node P.

$$t = \begin{cases} -\dfrac{\sqrt{r^2 - \tan^2\phi\left(l^2 - r^2\right)} - r}{\tan\phi} & \left(\dfrac{\pi}{2} < |\phi| < \pi\right) \\ \dfrac{\sqrt{r^2 - \tan^2\phi\left(l^2 - r^2\right)} - r}{\tan\phi} & \left(0 < |\phi| < \dfrac{\pi}{2}\right) \\ -\sqrt{l^2 - r^2} & \left(\phi = -\dfrac{\pi}{2}\right) \\ \sqrt{l^2 - r^2} & \left(\phi = \dfrac{\pi}{2}\right) \\ 0 & (\phi = 0) \end{cases} \tag{2.11}$$

As the result, the internal force $\overrightarrow{F_{Int}}$ in the inverse mapping is defined using t as follows.

$$\overrightarrow{F_{Ext}} = \left(t - \left|\overrightarrow{PP_1}\right|\right)\overrightarrow{N} \tag{2.12}$$

Chapter 3 Generation of SAIs with no insufficiency of approximation at depression

In this chapter, we solve the problem of insufficiency of approximation of semi-regular mesh to 3-dimensional shape model when generating SAI. For generating SAIs with no insufficiency of approximation at depression, we operate several steps for the mesh to get two features, local regularity and an approximation of surface of 3-dimensional model. First, we get the boundary of 3-dimensional shape model. Next, we construct a set of several shape datum called “guiding intermediate shapes” from the boundary of 3-dimensional shape model. The guiding intermediate shapes changes from sphere shape which is equal to initial mesh to the shape which is equal to the boundary of 3-dimensional shape model. Finally, by converging the mesh to guiding intermediate shapes from sphere shape to the shape which is equal to the boundary of 3-dimensional shape model, we get the semi-regular mesh which has two features of local regularity and the approximation of surface of 3-dimensional model. Moreover, by mapping geometrical attributes onto a sphere, we generate SAI of 3-dimensional shape model.
In this chapter, first, we describe a defect of former method when generating SAIs at depression. And, we describe the concept and abstract of proposed method against the problem. Next, as ready of constructing guiding intermediate shapes by our proposed method, we describe morphing closed curve with renewing energy function defined at nodes constructing closed curve step by step. And, we describe constructing guiding intermediate shapes and generating SAI using guiding intermediate shape. Finally, we conduct an experiment using artificial 3-dimensional shape model and show the advantage of our proposed method.

3.1 Defect of former method generating SAI at depression

With former method generating SAI at depression, we use one of the deformable surface methods that initial mesh which is sphere shape put around 3-dimensional shape model, calculate iteratively, deform the shape, and fitting onto the surface of 3-dimensional model. And, we try to construct semi-regular mesh which has two

features of local regularity and the approximation of surface of 3-dimensional model. Then, we generate SAI mapping geometrical attributes "simplex angle" onto a sphere surface.

When approximating 3-dimensional shape model onto mesh model at depression, we have a problem that mesh is not converged onto surface of 3-dimensional model because the mesh covers over the depression. As a result of this converging local minima, the mesh is satisfied with only local regularity, but is not satisfied with the approximation of surface of 3-dimensional model (See Fig. 3.1). Or, the mesh is satisfied with only the approximation of surface of 3-dimensional model, but is not satisfied with local regularity (See Fig. 3.2).

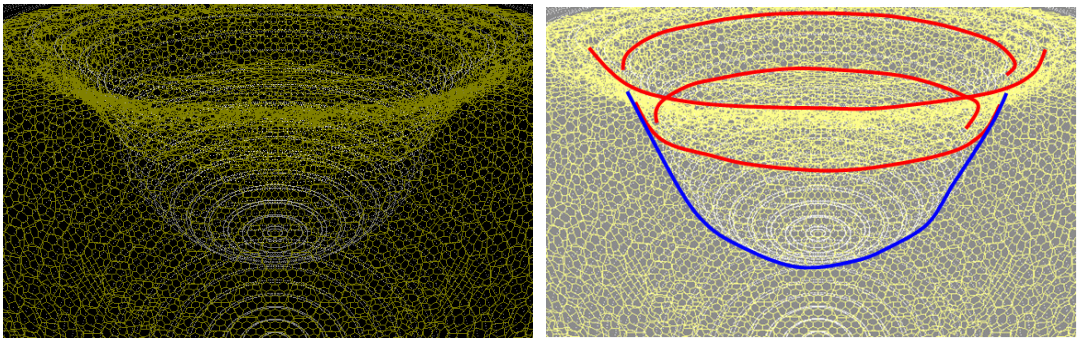


Figure 3. 1: Mesh with local regularity, and bad approximation of model surface

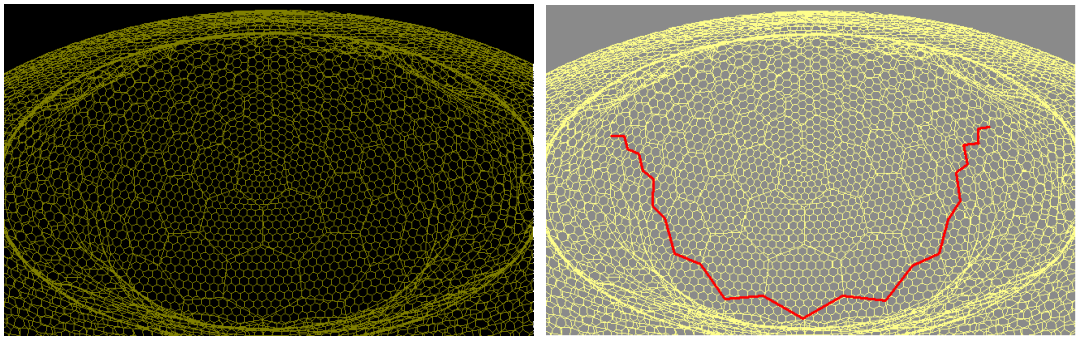


Figure 3. 2: Mesh with no local regularity, and good approximation of model surface

We regard the cause of this converging local minima as that we try for semi-regular mesh to acquire two features which are local regularity and the approximation of surface of 3-dimensional model with single process.

Then, we propose a new method against this converging local minima at depression.

Our new method is that mesh acquires two features of local regularity and the approximation of 3-dimensional shape model not with single process but with multi-process. With single process, the mesh can not obtain all information need for two features from 3-dimensional shape model. However, the mesh can obtain information need for one feature with one process. Then, the mesh can obtain all information need for all features with multi-process. Note that we perform two processes in this thesis, because the mesh need two features.

With first process, we generate the deformation series of 3-dimensional shape model without considering local regularity. With the second process, we obtain the approximation of 3-dimensional shape model which is satisfied with local regularity using the series. Concretely speaking about these two processes, (1) we construct a set of model datum called "guiding intermediate shapes" which shape changes from the shape of initial mesh to the boundary of 3-dimensional shape model. Note that guiding intermediate shapes have the feature of the approximation of 3-dimensional shape model., and (2) we converge semi-regular mesh onto guiding intermediate shapes one by one. And, we obtain for semi-regular mesh two features that are local regularity and the approximation of surface of 3-dimensional model. We use these two processes for not happening to the mesh converging local minima.

3.2 Overview of Generating “guiding intermediate shapes”

In this section, we describe the overview of generating “guiding intermediate shapes”. To generate guiding intermediate shapes, we first cut a series of proper sections. Second, we morph each section to a circle. Finally, we unify the results of morphing of all sections, and generate a series of 3-dimensional shape models which are guiding intermediate shapes.

3.3 Morphing closed curve by gradual renewing of energy function

In this thesis, for constructing guiding intermediate shapes, we morph closed curve which is expressed the list of line segments. Energy function for converging closed curve is defined at a node defined at the end of a line segment.

In this section, for example of morphing something converging by energy function, we consider closed curve defined energy function at each node of line segments. We morph closed curve from initial shape of closed curve to circular shape (sphere shape is expressed by regular polygon.) gradually by converging energy function to maximum or minimum. The detail definition of energy function is described at Section 3.3.3.

3.3.1 Morphing by converging calculation with repeating same number of times

We morph closed curve by converging calculation with repeating same number of times. Figure 3.3 shows the graph of energy function at converging calculation. The sum of repeating number of times is about 57000. The converging circumstances (See Fig. 3.4) every 5700 repeating number of times does not move constantly, and the result of morphing closed curve is not good. Unless we morph closed curve moving constantly, the distance of closed curve moving is long. The guiding intermediate shapes describing detail as follows are a set of models, and need to change the shape gradually. Because of this reason, this result is not satisfied with our proposed method generating SAIs at depression. Then, the moving distance of closed curve must be constant.

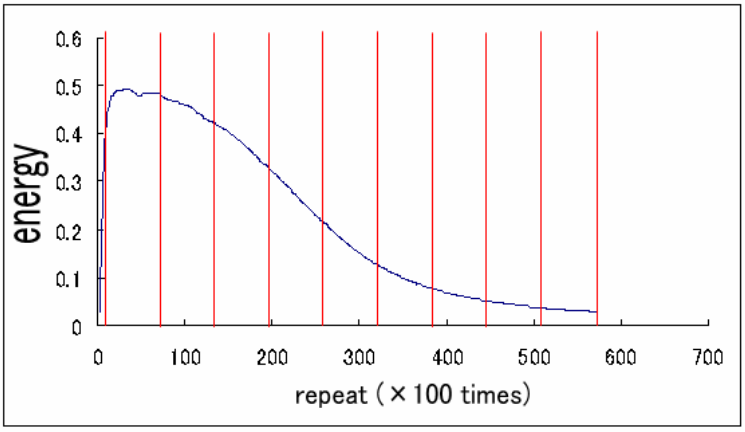


Figure 3. 3: Graph of repeating number of times and energy with repeating same number of times

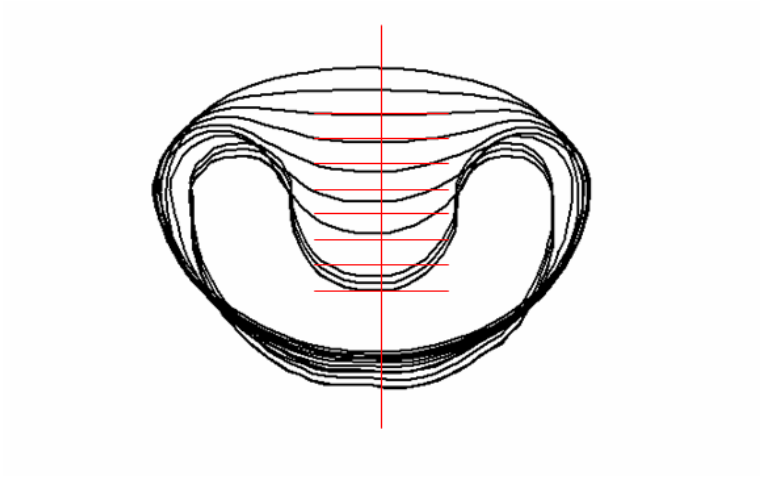


Figure 3. 4: Morphing by converging calculation with repeating same number of times

3.3.2 Morphing by converging calculation with renewing energy function

The method of morphing, by converging calculation with repeating same number of times, can not be realized morphing with closed curve constant moving. Then, we realize morphing with closed curve changing the shape constantly, by converging calculation with renewing energy function defined at each node of line segments.

We morph closed curve by converging calculation with renewing energy function. Energy function is sum of two kinds of energy functions. One is the energy function called the “angle energy function” which is defined from angle between a node and both neighbor nodes. Another is the energy function called the “distance energy function “ which is defined from distance between two neighbor nodes. In this thesis, we assume that each distance between two neighbor nodes is constant. Because of this assumption, we don't renew distance energy function. In this section, we pick up angle energy function.

(1) Renewing angle energy function

The angle energy function is defined error between present angle and the angle at time of converging between a node and its neighbor nodes. Moreover, the angle energy function has function of closing present angle to the angle at time of converging. The

larger is the error, the larger is the angle energy function. The smaller is the error, the smaller is the angle energy function. Then, the angle at the time of converging is renewed by linear transform from the initial angle to the angle at the circle shape (the angle is equal to an interior angle of regular polygon). By this renewing the angle at the time of converging, we renew the angle energy function.

(2) Results of morphing

Figure 3.5 is a graph of repeat number of times and energy at converging calculation with renewing energy function. The beginning of repeat calculation shows energy decreasing and it shows converging by first energy function. Next, at the same time of renewing energy function, energy increases. Then, new converging is performed by new energy function. And so on, by repeating of renewing energy function and converging, we realize morphing with constant moving distance of closed curve (See Fig. 3.6).

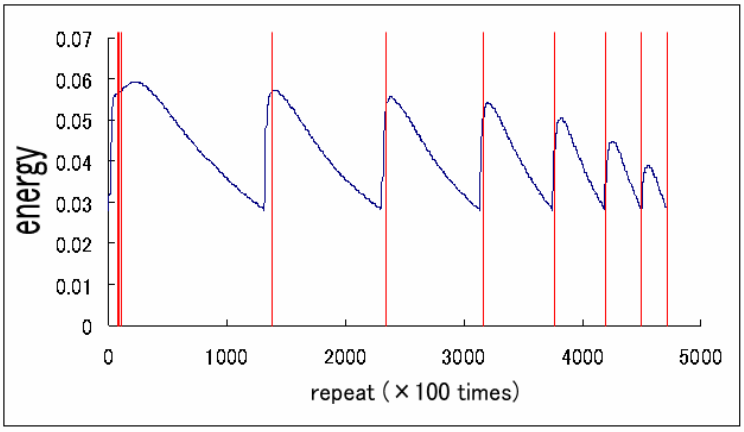


Figure 3. 5: Graph of repeat number of times and energy with renewing angle energy function

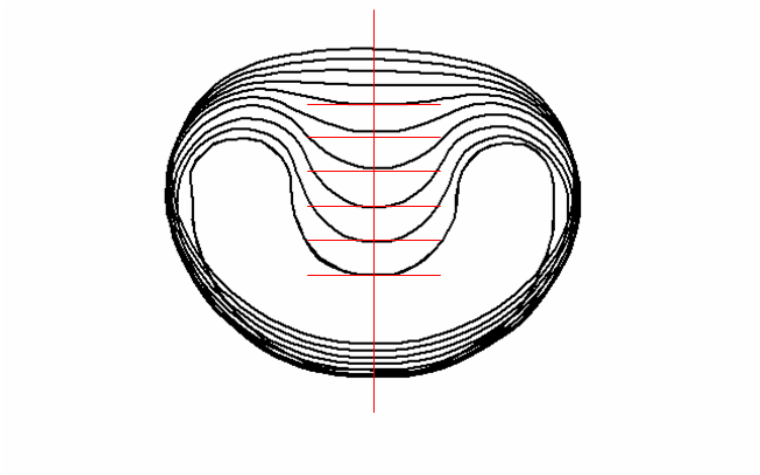


Figure 3. 6: Morphing by converging calculation with renewing energy function

3.4 Generation of SAIs using guiding intermediate shapes

In this section, we describe the detail about guiding intermediate shapes, and how to converge the mesh using guiding intermediate shapes. And, we describe how to construct guiding intermediate shapes.

3.4.1 The guiding intermediate shapes

The guiding intermediate shapes consist of a set of points. And, the guiding intermediate shapes guide mesh which initial shape is sphere (See Fig. 3.7) onto 3-dimensional shape model (See Fig. 3.8). The guiding intermediate shapes are a sequence of several shapes changing from sphere shape to the shape of the approximation of 3-dimensional shape model (See Fig. 3.9).

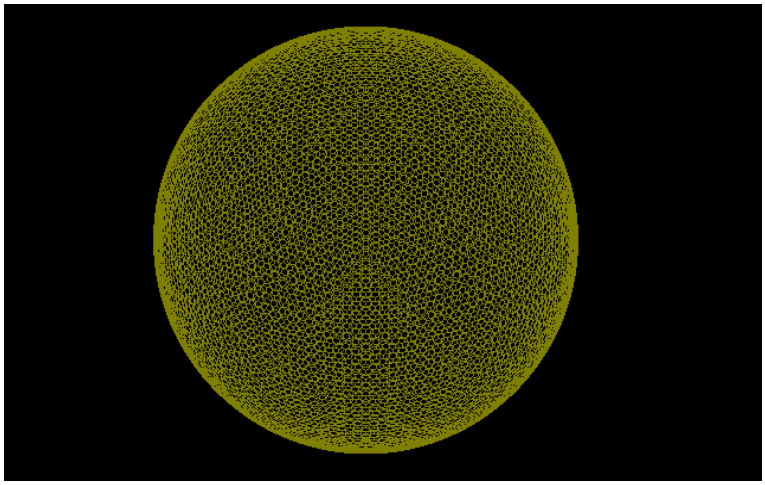


Figure 3. 7: The mesh which initial shape is sphere

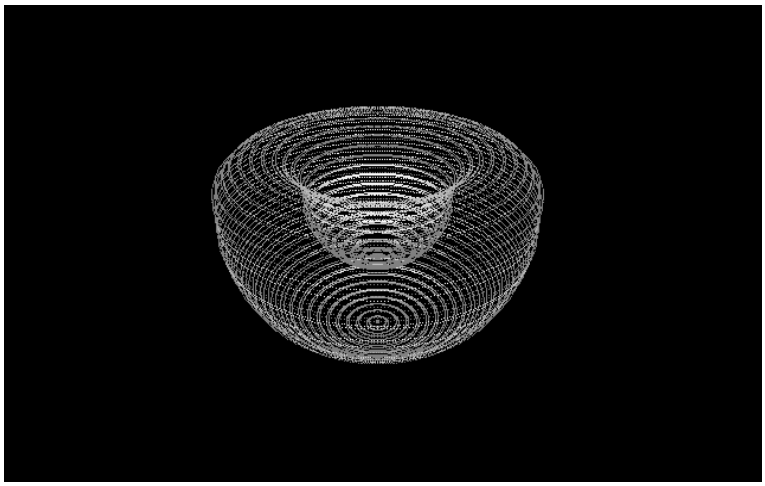


Figure 3. 8: 3-dimensional shape model

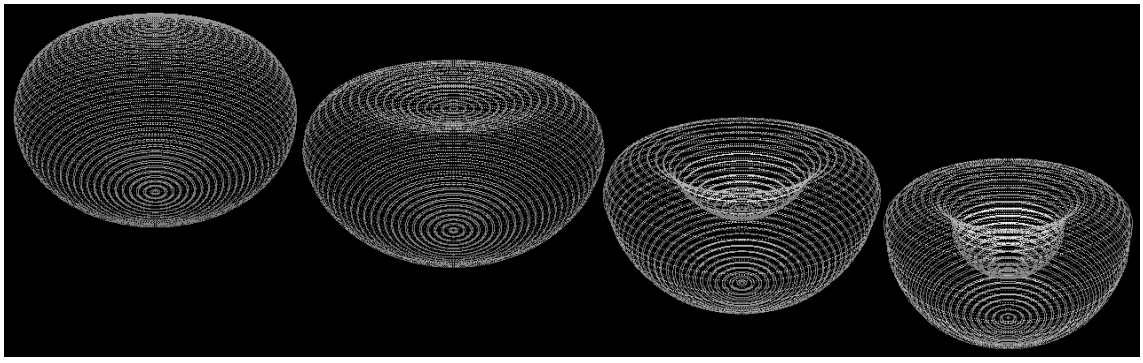


Figure 3. 9: The guiding intermediate shapes

3.4.2 Converging the mesh using guiding intermediate shapes

We converge the mesh whose initial shape is sphere onto a guiding intermediate shape whose shape is near by the shape of initial mesh (See Fig. 3.10). We repeat converging the mesh several times renewing the guiding intermediate shape step by step, using the original SAI method. Finally, we converge the mesh onto a set of points which consist of 3-dimensional shape model.

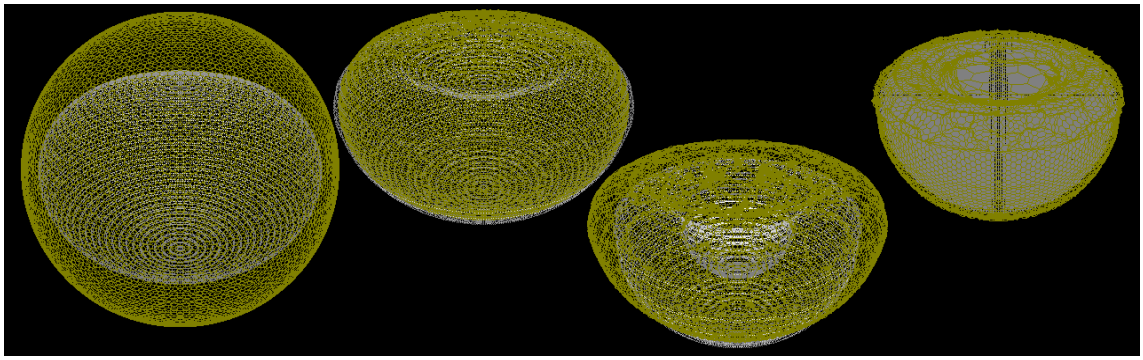


Figure 3. 10: Circumstances of converging the mesh using guiding intermediate shapes

3.4.3 How to construct the guiding intermediate shapes

In this section, we describe how to construct the guiding intermediate shapes. Various methods how to construct the guiding intermediate shapes are considered. Then, we consider the method to construct 3-dimensional guiding intermediate shapes by merging several 2-dimensional datum which express sections of 3-dimensional shape model.

First, we do sampling sections of 3-dimensional shape model. And, we gradually morph the sections to the sections of the initial mesh which is the shape of circle by gradually renewing energy function that we describe its details in Section 3.3. Finally, we merge each result of morphed sections, and the results of merging sections are the guiding intermediate shapes. We describe the method and detail of constructing the guiding intermediate shapes as follows.

(1) Sampling outline of 3-dimensional shape model

We do sampling the outline of 3-dimensional shape model. In this chapter, because of artificially generating 3-dimensional shape model, it is easy sampling the outline of 3-dimensional shape model. However, in the case of obtaining 3-dimensional shape model by measuring a real sample, we take advantage of the fact that we describe following. Because the data which is obtained by measuring a real sample is expressed a set of triangle patches, a set of triangle patches' boundaries which cross over the section expresses the outline of 3-dimensional shape model.

(2) Morphing the outline of 3-dimensional shape model

We morph the outline of 3-dimensional shape model to the shape of initial mesh which is circle. And, the results of this morphing the outlines are merged at all sections. We reverse this results. Then, they are the guiding intermediate shapes. About how to change the outline to circle shape gradually, we take advantage of repeat calculation by gradually renewing energy function whose detail is described in Section 3.3.

Moreover, this energy function consist of two kind of energy functions, one is the angle energy function defined with the angle between a node and both neighbor nodes, another is the distance energy function defined with the distance between two neighbor nodes.

(2.1) The angle energy function

We define two vectors between a node and both neighbor nodes (See Fig. 3.11). Let the angle between two vectors be $\theta[\mathrm{rad}](-\pi < \theta \le \pi)$. Then, the angle when energy is minimum at converging a node is $\theta_0[\mathrm{rad}](-\pi < \theta \le \pi)$. The error between present angle and the angle when energy is minimum is expressed the error ratio as Equation (3.1).

$$rate_{\mathrm{angle}} = \frac{\theta - \theta_0}{2\pi} \quad \left(-1 < rate_{\mathrm{angle}} < 1\right) \tag{3.1}$$

Using the error ratio, the angle energy function is defined as follows (See Fig. 3.12).

$$\mathrm{E}_{\mathrm{angle}} = rate_{\mathrm{angle}}{}^{2} \tag{3.2}$$

The angle energy function E_{angle} takes minimum value at $\theta = \theta_0$ (See Fig. 3.11).

At each morphing process, the angle energy function is renewed every repeat calculation of converging the mesh. How to renew the angle energy function is changing the angle θ_0 when the energy is minimum at every morphing process.

We use the linear transform between the angle θ_{start} at initial condition of the outline and the angle $\theta_{goal}(=\pi-(\text{interior angle of regular polygon}))$ of regular polygon whose shape is near by the shape of circle. And, we renew the angle θ_0 as follows.

We assume generating N morphing images. At generating n th morphing image, the angle energy function is defined by $t = \frac{n}{N}$ as Equation (3.3). Like normal repeat calculation, we repeat converging calculation until the sum of all energy functions which are the angle energy function and the distance energy function is less than a threshold. After converging calculation, the angle energy function is renewed by $t = \frac{n+1}{N}$ for generating $n+1$ th morphing image.

$$\begin{aligned} t &= \frac{n}{N} \\ \theta_0 &= (1-t)\theta_{start} + t\theta_{goal} \quad (0 \le t \le 1) \end{aligned} \tag{3.3}$$

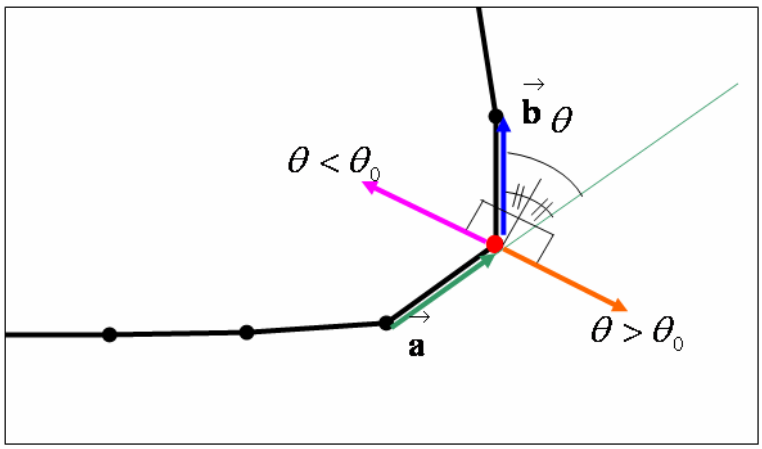


Figure 3. 11: The angle energy affecting between a node and both two neighbor nodes

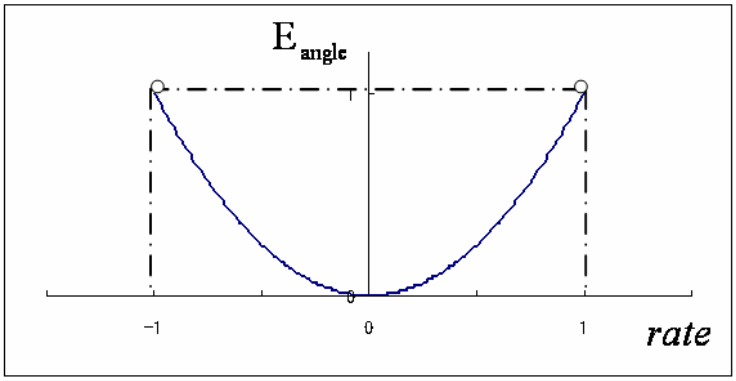


Figure 3. 12: Graph of the angle energy function

(2.2) The distance energy function

We consider the distance d between two neighbor nodes (See Fig. 3.13). Moreover, we define the distance d_0 when the energy is minimum at converging calculation. The error is expressed by the error ratio between present distance d and the distance d_0 which is the distance when the energy is minimum as Equation (3.4).

$$rate_{\text{dist}} = \frac{d - d_0}{d_0} \quad \left(-1 < rate_{\text{dist}} < +\infty\right) \tag{3.4}$$

Using this error ratio, we define the distance energy function as Equation (3.5). We consider Equation (3.5) as a spring model (See Fig. 3.14).

$$\mathrm{E}_{\text{dist}} = rate_{\text{dist}}^{\ 2} \tag{3.5}$$

At morphing the outline, we renew the distance energy function every repeat calculation of converging. How to renew the distance energy function is that we renew the distance d_0 when the energy is minimum as follows.

We renew the distance d_0 using the linear transformation between the distance d_{start} at initial outline condition and the distance d_{goal} at final outline condition as Equation (3.6).

About the angle energy function, we assume that we generate N morphing images. When we generate nth morphing image, we define the distance energy function by $t = \frac{n}{N}$. Like normal repeat calculation, until the sum of all energy functions which are

the angle energy function and the distance energy function is less than a threshold, we repeat converging calculation. When the converging calculation is finished, the distance energy function is renewed as $t = \frac{n+1}{N}$ for generating $n+1$th morphing image.

$$t = \frac{n}{N}$$
$$d_0 = (1-t)d_{\text{start}} + td_{\text{goal}} \quad (0 \le t \le 1) \tag{3.6}$$

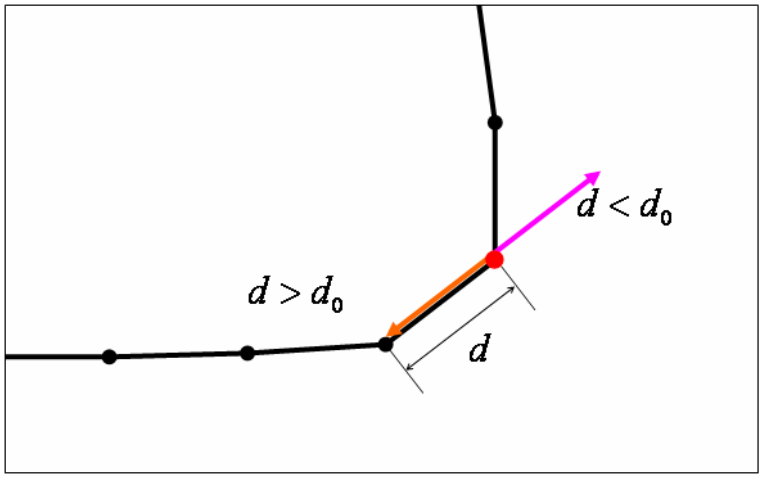


Figure 3. 13: The distance energy affecting between two neighbor nodes

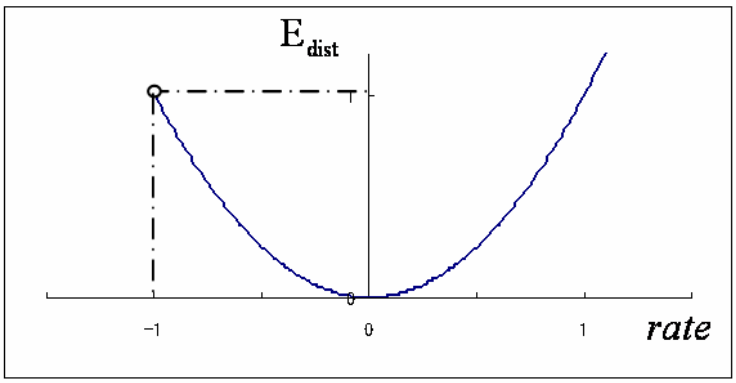


Figure 3. 14: Graph of the distance energy function

(2.3) The energy defined at each node

At each node, one angle energy function and two distance energy functions are defined. We define the sum of all energy functions as the energy at each node as Equation (3.7). Note that β is the ratio between the angle energy function and the distance energy function. (In this thesis, let it be $\beta = 1$.)

$$E_{node} = E_{angle} + \beta E_{dist1} + \beta E_{dist2} \tag{3.7}$$

The sum of all energy functions at all nodes is defined as follow equation.

$$E_{all} = \sum E_{node} \tag{3.8}$$

Moreover, the threshold $E_{threshold}$ which a threshold used for stopping repeat calculation is defined as follow equation using the error $error(\%)$ about the angle energy function and the distance energy function.

$$\begin{aligned} &error = 1\% \\ &E'_{angle} = error^2 \\ &E'_{dist} = error^2 \\ &E'_{node} = E'_{angle} + \beta E'_{dist1} + \beta E'_{dist2} \\ &E_{threshold} = \sum E'_{node} \end{aligned} \tag{3.9}$$

(3) The affine transformation for merging morphed closed curves for making guiding intermediate shapes

The morphed closed curve is decided by repeating calculation of gradually renewing each energy function. This morphing process can only determine shape of closed curve and that is not functional to determine its proper location and orientation at all. Then, the center of gravity of the closed curve is located at random. And, it is probable that the closed curve itself rotates. Moreover, there is no consideration about the scaling of the closed curve, because of morphing the outline at constant distance between two neighbor nodes.

We need to determine its location and orientation properly.

In this section, we describe the detail of the affine transformation: translation, rotation, scaling which are done to a set of the closed curves changing the shape from the boundary of 3-dimensional shape model's section to the shape of circle.

(3.1) Location

The proper location of the closed curve is decided in obedience to a set of the closed curves obtained by morphing. The closed curve expresses the section boundary of 3-dimensional shape model. However, the center of gravity of the closed curve is expresses the center of gravity of the section of sphere which is initial semi-regular mesh shape. Here, the proper location of the center of gravity of a set of the closed curves is decided by the linear transforming between the center of gravity of the section of 3-dimensional shape model boundary.

To modify the location of the closed curve, first, we obtain the location of the center of gravity of the closed curve which we modify the location. Next, we properly determine where the center of gravity of the closed curve is. Finally, we transform the closed curve by moving the distance between the present location and the proper location.

(3.2) Rotation

In this section, first, we define the orientation of a closed curve angle. orientation is the difference between in some coordinate system (e.g. object-oriented coordinate system) and in the world coordinate system. Next, we describe how to obtain the proper angle of each closed curve from initial closed curve. Finally, we describe how to rotate the closed curve.

First, we define the angle of the closed curve. Figure 3.15 shows that the closed curve is expressed by list of n line segments. Here, we assume the length of each line constant. Now, the points of each line segment end are called nodes, and they are defined at the node number $0,1,\cdots,n-1$.

Here, we locate the regular polygon whose node number is n around the closed curve. Each center of gravity (the closed curve and the regular polygon) is the same position. Moreover, each node of the regular polygon is defined the node number $0,1,\cdots,n-1$. We define the center of gravity of the closed curve and the regular polygon as the origin. And, we define the line from the origin to any direction as the axis line. The direction of the axis line is constant at morphing all closed curves.

Next, we rotate the regular polygon around the origin. We define vectors from each

node of the regular polygon to the node with the same number on the morphed curve. We rotate the regular polygon when the sum of the vectors' lengths becomes minimum. We determine the orientation of the closed curve from the angle θ_0 between the axis line of regular polygon and the line of the closed curve when the sum of the vectors' lengths is minimum (See Fig. 3.16).

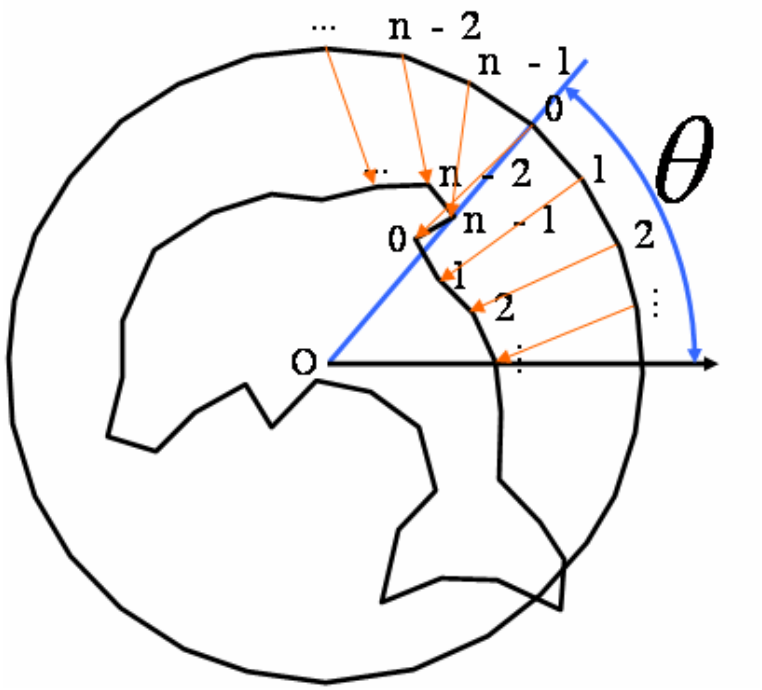


Figure 3. 15: The vectors from the node of the regular polygon to the node of the closed curve

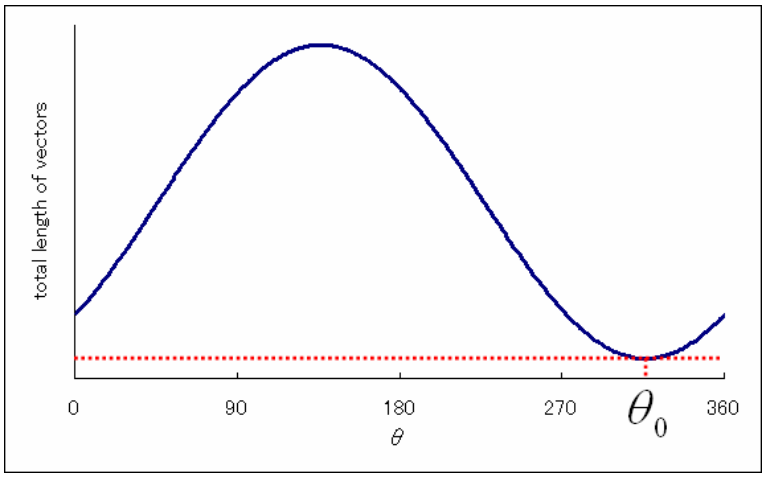


Figure 3. 16: Graph of the angle of the closed curve and the sum of vectors' lengths

Next, we describe how to obtain the proper angle of the closed curve from the initial closed curve. The orientation of the closed curve is uniquely determined at each morphing. Then the proper angle of the closed curve is the same of the initial closed curve's angle before morphing.

Finally, about a set of the closed curve obtained by morphing, we obtain the angle of each closed curve of which rotation we modify. Then we rotate each closed curve in proportion to the error between the present angle and the proper angle. Then, we can modify the angle of a set of the closed curves.

(3.3) Scaling

The scaling of the guiding intermediate shapes changes from the same scale of the initial mesh to the same scale of the section of 3-dimensional shape model boundary. Then, it is necessary to modify the scaling of all closed curves. The scaling of the closed curve changes from the same scale of the initial mesh to the same scale of the section of 3-dimensional shape model boundary. Here, we take advantage of the fact that the distance between two neighbor nodes is constant so as to scale the closed curve.

Here, we define a proper distance between two neighbor nodes. The proper distance is linear transformation between the distance at closed curve's approximation of the section of 3-dimensional shape model boundary and the distance at closed curve's approximation of the section of the initial mesh boundary which is the shape of circle.

To modify the scaling of the closed curve, we obtain the present distance between two neighbor nodes of the closed curve which we modify the distance of. And we obtain the proper distance between the neighbor nodes of the closed curve. Then, we scale the closed curve at the error ratio between the present distance and the proper distance.

3.4.4 Merging the closed curves

Finally, we merge the closed curves morphed from several outline extracted from 3-dimensional shape model. We reverse the merging results, and the results are the guiding intermediate shapes.

3.5 Experiment

We have an experiment of converging mesh using our proposed method. In this section, we describe this experiment in detail.

3.5.1 Detail of the experiment and about the data

We converge the closed surface by our proposed method and former method. And we compare these two method in view of accuracy. We compare the degree of converging the proper surface. And, we consider the caution about the comparison of the results of these two methods. Moreover, we compare calculation time and consider about both two methods.

We use initial mesh which divide the dodecahedron five times (See Fig. 3.17). The number of faces of the mesh is $20 \times 4^5 = 20{,}480$. And, we use artificial 3-dimensional shape model (See Fig. 3.18). We generate ten guiding intermediate shapes from the sphere shape to the boundary of 3-dimensional shape model (See Fig. 3.19).

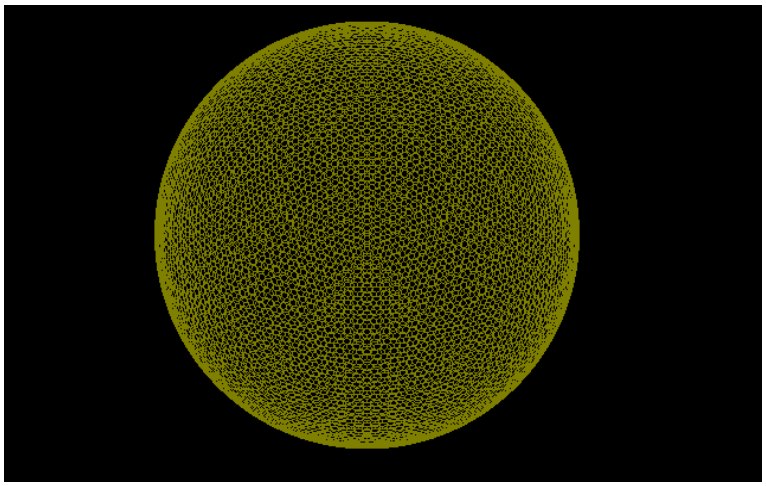


Figure 3. 17: Initial mesh

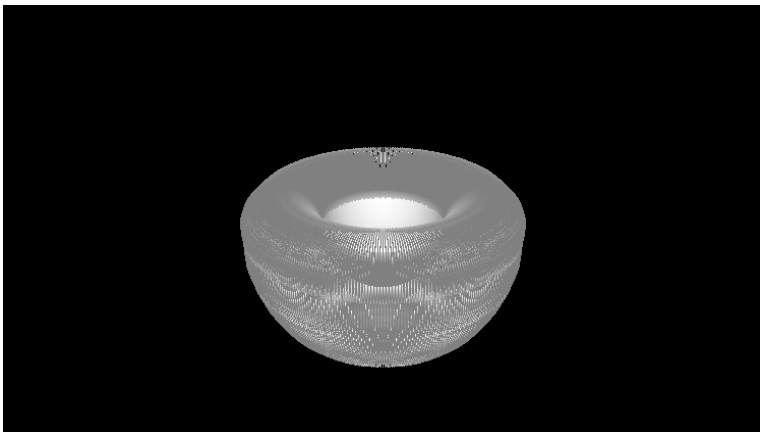
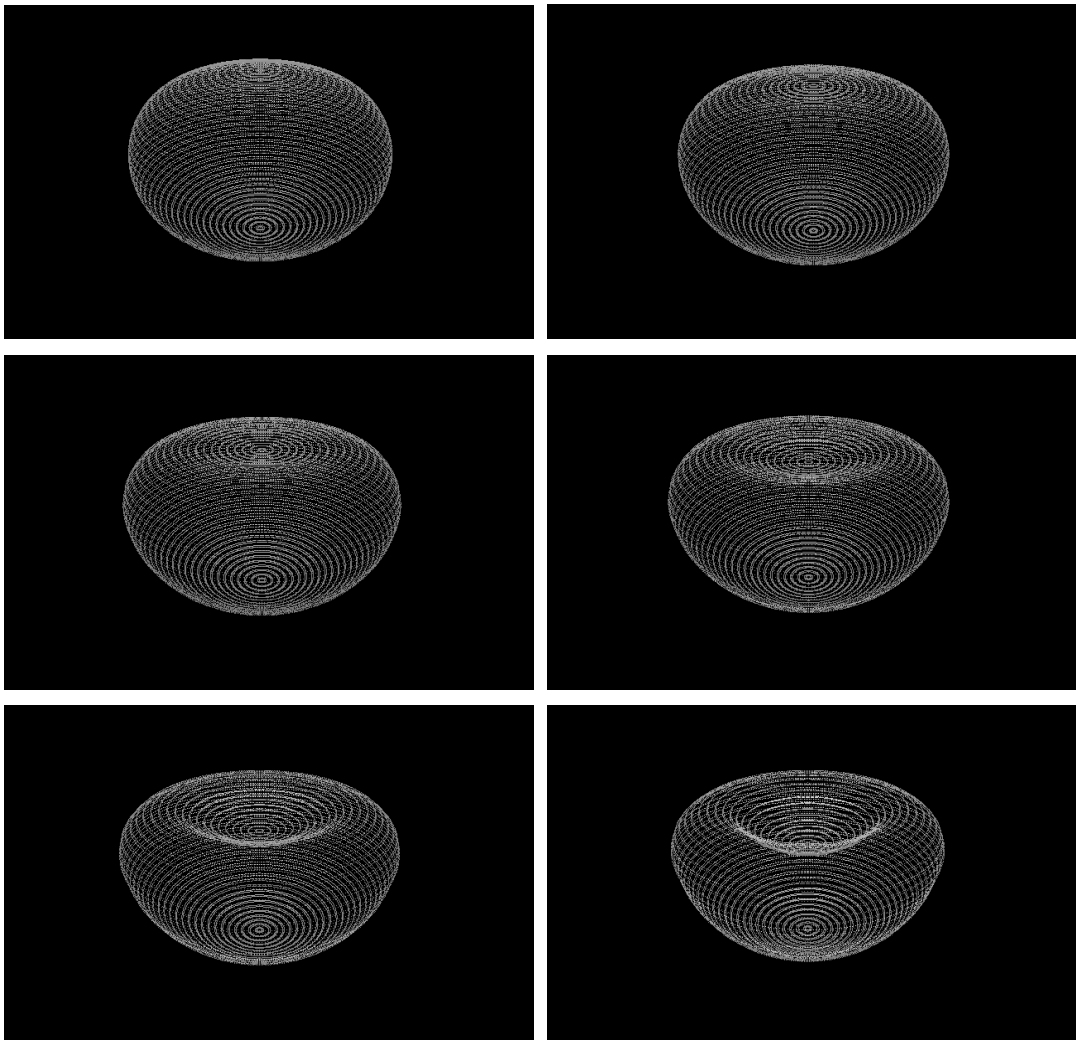


Figure 3. 18: The artificial 3-dimensional shape model



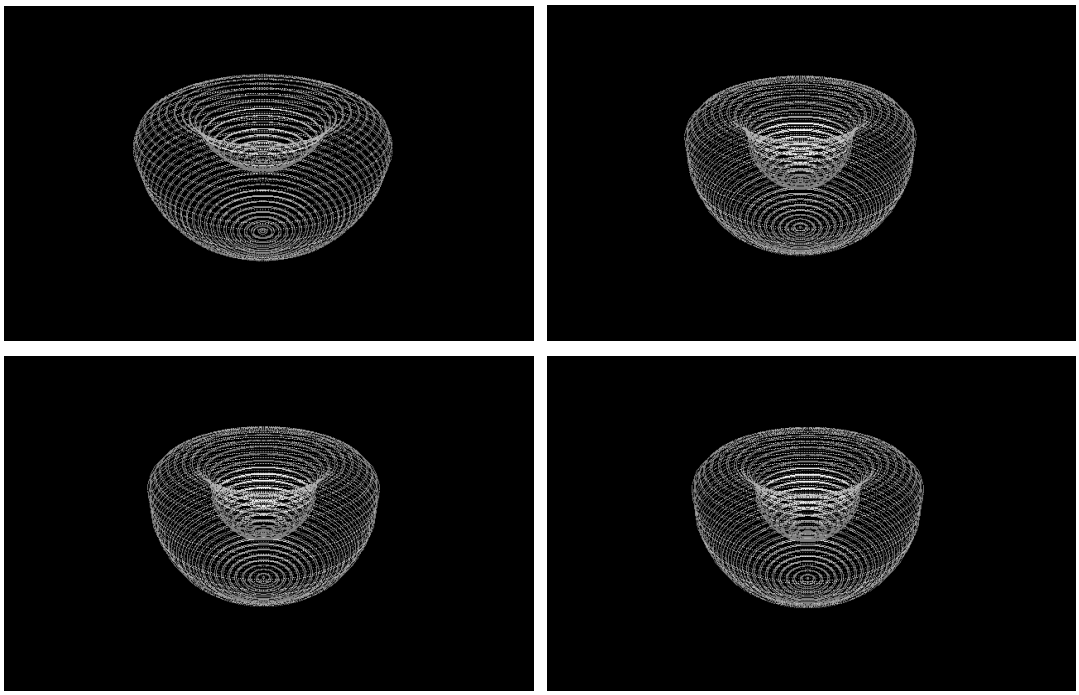


Figure 3. 19: 10 guiding intermediate shapes

3.5.2 Result of the experiment

We converge the mesh by our proposed method using the guiding intermediate shapes (See Fig. 3.20) and the former method (See Fig. 3.21, 3.22). We define an approximation error of a node as the distance between the node and the closest point on the proper surface. The error is visible mapping onto the mesh at the rate of converging local minima scaling 10 times (See Fig. 3.23).

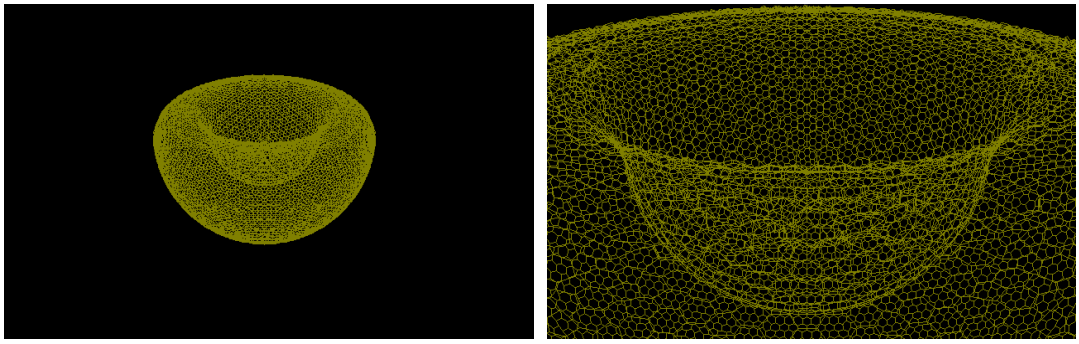


Figure 3. 20: The result of converging using guiding intermediate shapes and its macrograph

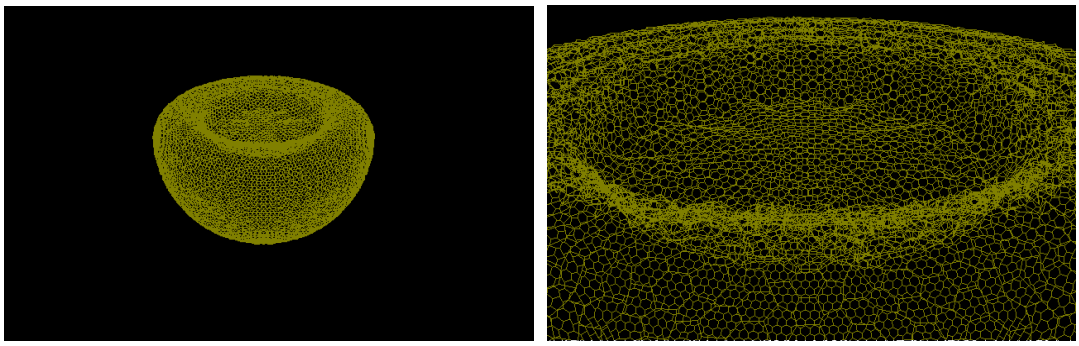


Figure 3. 21: The result of bad boundary approximation by former method and its macrograph

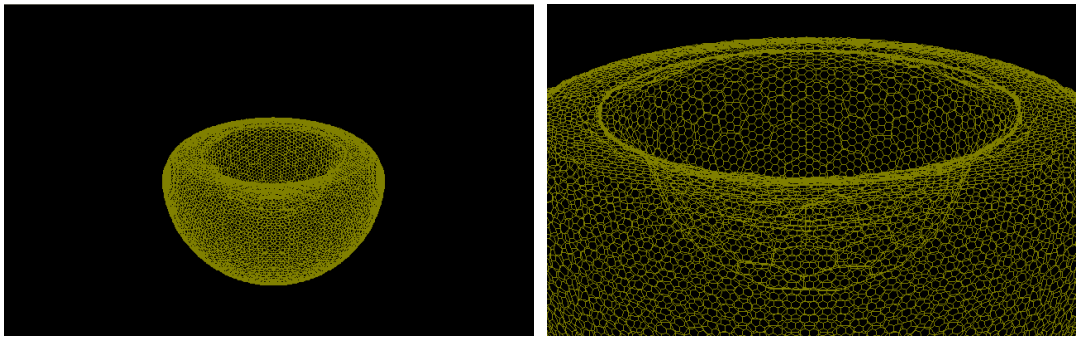


Figure 3. 22: The result of unsatisfying local regularity by former method and its macrograph

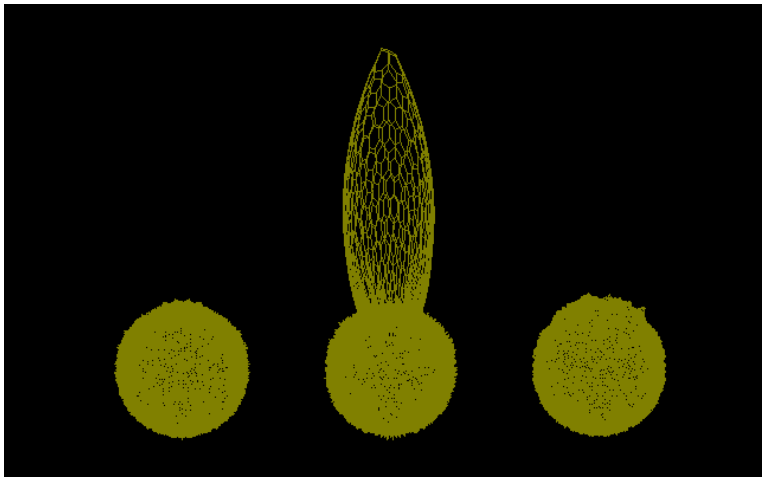


Figure 3. 23: The comparison of the converging result, left: using the guiding intermediate shapes, center: the former method (bad boundary approximation), right: the former method (unsatisfying local regularity)

Here, by the method using guiding intermediate shape, the mesh has good approximation of artificial 3-dimensional shape model boundary. However, by the former method, the mesh has bad approximation of the boundary.

The length between two neighbor nodes of the mesh by both methods is Table 3.1. Both of them, there is equal order of variance and nearly equal value of mean. Then, we regard the conditions of converging the mesh by both methods as the same.

Moreover, by the former method (unsatisfied local regularity), the mesh is converged onto depression at the center of the artificial 3-dimensional shape model (See Fig. 3.22). However, the length of the mesh by the former method (unsatisfied local regularity) is not the same order with the one by our proposed method using guiding intermediate shapes (See Table 3.1). Moreover, Figure 3.22 shows that by this method the mesh unsatisfied the feature of local regularity.

Table 3. 1: The comparison about the length between two neighbor nodes of the mesh

	variance	mean	$\frac{\text{variance}}{\text{mean}}(\%)$
proposed method	5.411×10^{-5}	1.079×10^{-3}	5.015(%)
former method: wrong approximation of the surface area	1.767×10^{-5}	1.105×10^{-3}	1.599(%)
former method: unsatisfied with the local regularity	1.426×10^{-4}	1.048×10^{-3}	13.61(%)

3.5.3 Discussion

At repeat calculation of converging the mesh, two kinds of force affect the mesh. One is the force that the mesh converge onto the surface, the other is the force that each length between two neighbor nodes of the mesh keeps constant. Here, we assume the mesh converging situation according to the section of 3-dimensional shape model. At converging the mesh by the former method, there are two kinds of areas (See Fig. 3.24). One is the area converging surface early. Another is the area converging surface late.

First, the length between two neighbor nodes at the area converged onto the surface early time changes shorter and shorter. At the same time, the length at the area converged onto the surface late time changes shorter and shorter. Then, the mesh is insufficient at the area converged late time. However, the mesh at blue color area is already converged and the energy is balanced. Then, the mesh at the converging area hardly moves. As a result of this, (1) in the case of the force that the mesh is converged onto the surface is stronger than the force keeping the length of the mesh, the length at only the area converged onto the surface late time changes longer and longer. The mesh has good approximation of 3-dimensional shape model boundary. But, The mesh does not satisfy local regularity (See Fig. 3.25). Or, (2) in the case of the force that the mesh is converged onto the surface is weaker than the force keeping the length of the mesh, The length at the area converged onto the surface late time does not change. The mesh satisfies local regularity. But, the mesh has bad approximation of 3-dimensional shape model boundary (See Fig. 3.26). As a result of this, it is difficult that the mesh has two features of local regularity and good approximation of 3-dimensional shape model boundary at the same time.

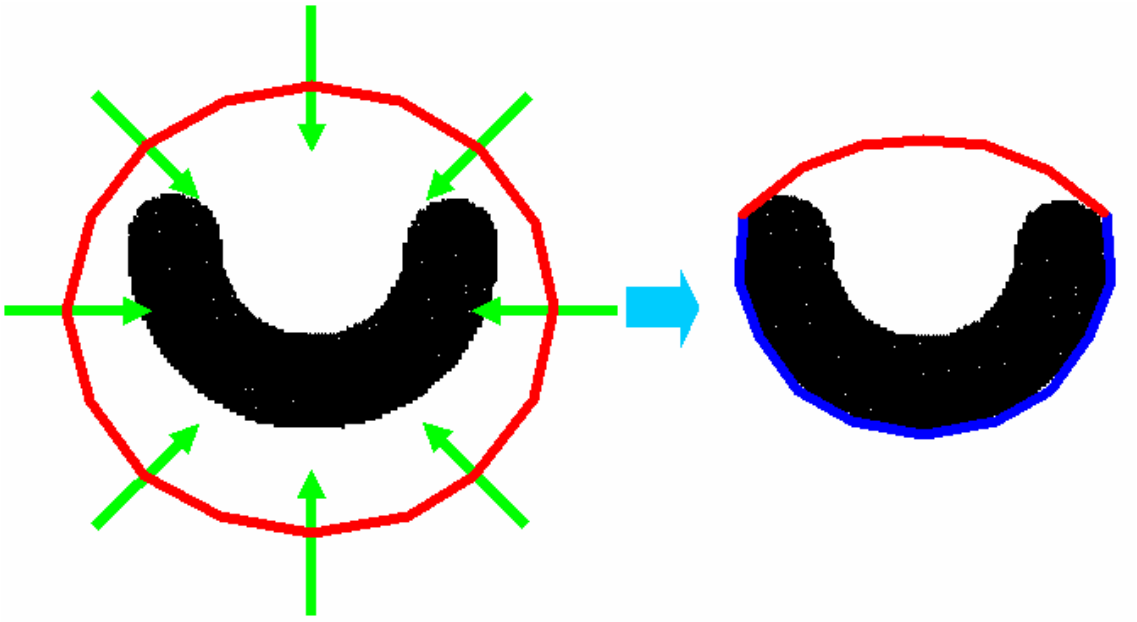


Figure 3. 24: The section of converging the mesh onto 3-dimensional shape model boundary by the former method

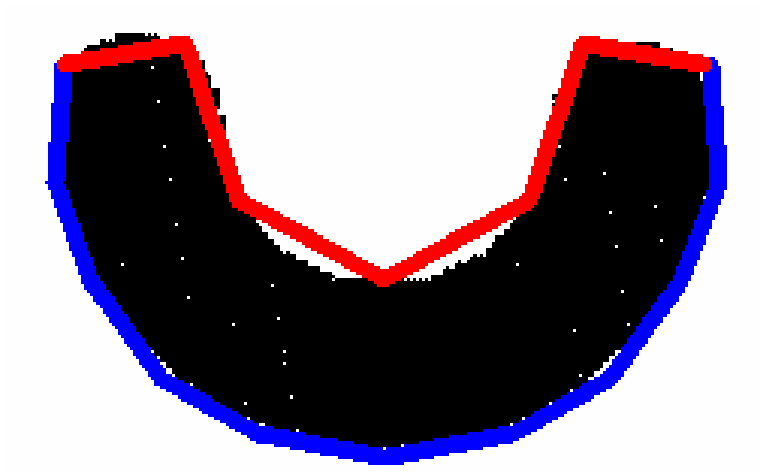


Figure 3. 25: In the case of unsatisfying local regularity

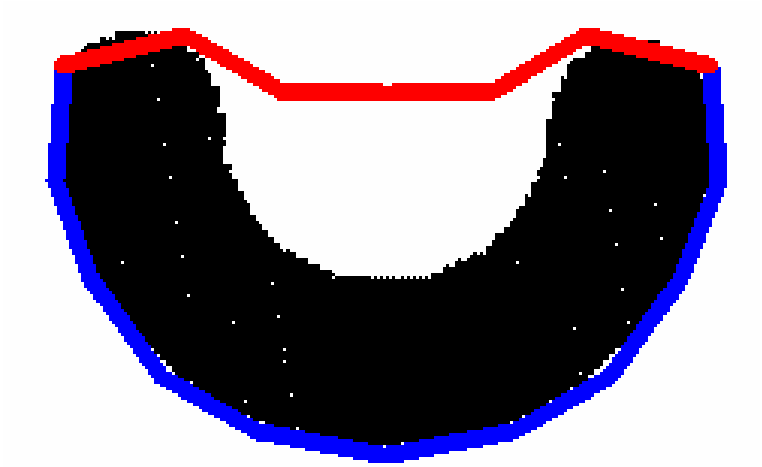


Figure 3. 26: In the case of bad approximation of 3-dimensional shape model boundary

On the other hand, by our proposed method using guiding intermediate shapes, the mesh is converged onto the surface of 3-dimensional shape model keeping balance. In this reason, it is capable of the balanced converging the mesh (See Fig. 3.27).

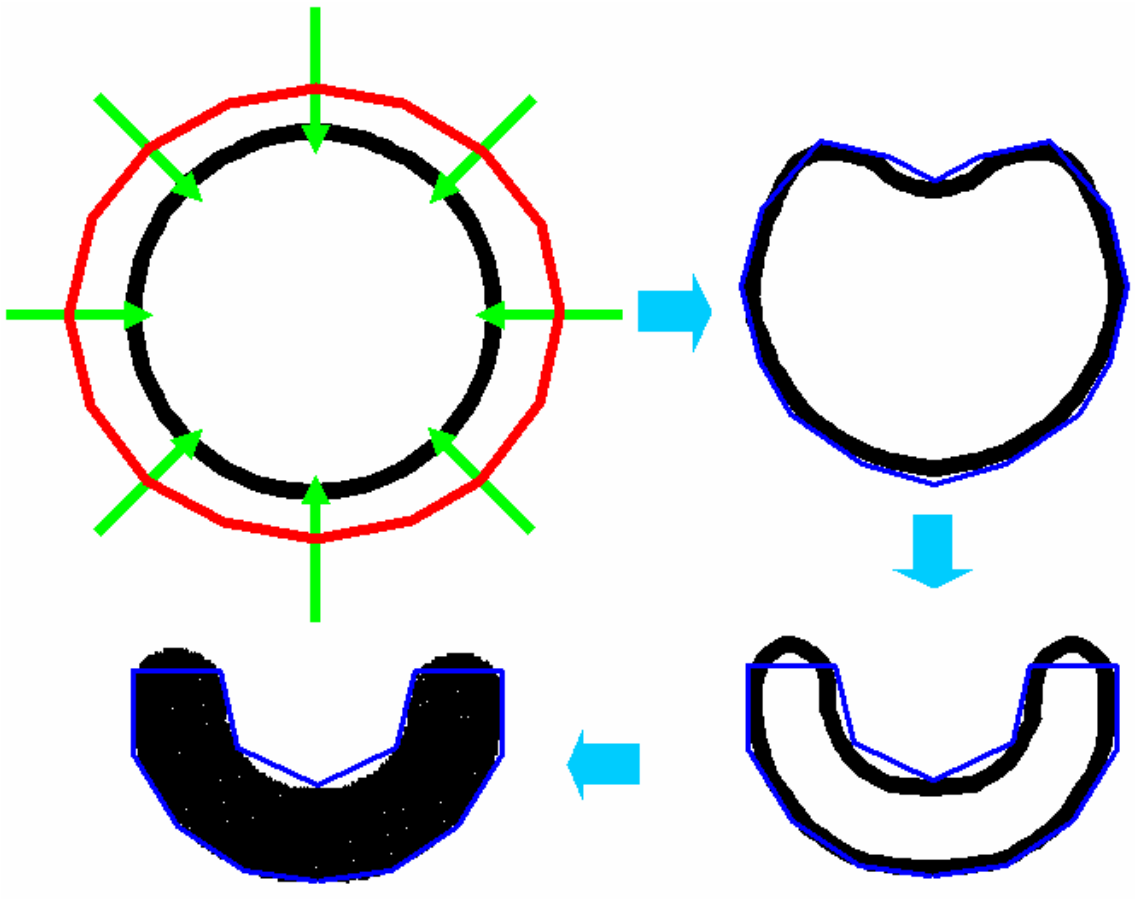


Figure 3. 27: Converging the mesh by the method using guiding intermediate shapes

3.5.4 The comparison of converging runtime and the consideration

Table 3.2 shows the comparison of converging runtime by the proposed method and the former method. The numbers of iteration are the same 5500 times by both methods. The proposed method is faster than the former method.

Table 3. 2: The comparison of run-time

	the total run time	the run time of one loop
proposed method	3976(sec) (1.1(h))	0.7229(sec)
former method	23732(sec) (6.6(h))	4.315(sec)

The reason why our proposed method is faster than the former method is because of the difference between the number of points constructing 3-dimensional shape model and the number of points constructing the guiding intermediate shapes (See Table 3.3). We calculate the energy defined with the mesh when we converge the mesh onto the surface of 3-dimensional shape model. In this energy calculation, we perform nearest neighbor nodes searching. In nearest neighbor nodes searching, the less points are there, the less time we spend.

Table 3. 3: The comparison of the number of points constructing each model

	points
3-dimensional shape model	286,977
middle shape data	33,840

Chapter 4 Analysis using SAIs

In this chapter, we perform 3D-object-shape analysis taking advantage of the similarity between their SAIs. We try to illustrate the result using a dendrogram.
First, we calculate the distance between each pair of 3-dimensional shape models. Note that we regard the distance as similarity between each pair of 3-dimensional shape models. Next, we construct the dendrogram by hierarchical cluster analysis with the distance between each pair of SAIs.
In this chapter, first, we describe how to calculate the distance between two 3-dimensional shape model. Next, we describe about hierarchical cluster analysis. Moreover, we describe about fast calculation of the distance between two 3-dimensional shape models. Finally, we perform an experiment of analysis.

4.1 The distance between two models defined by SAIs

In this section, we describe how to calculate the distance between two 3-dimensional shape models taking advantage of their SAIs. In each SAI, we first calculate the line which goes through both the origin and the feature node. Then we rotate the two SAIs which we would like to compare so as to align the two lines (See Fig. 4.1). Next, one SAI rotates around the axis little by little, and we calculate the difference from another SAI. After one SAI turn one revolution, we define the minimum difference between two SAIs as the distance between two 3-dimensional shape models.
In this section, first, we describe how to define the difference between two SAIs and calculate the difference. Next, we define the distance between two 3-dimensional shape models using the difference between two SAIs.
Note that the difference between the two SAIs depends on the configuration of their coordinate systems but the distance between the two models can be uniquely decided using the differences.

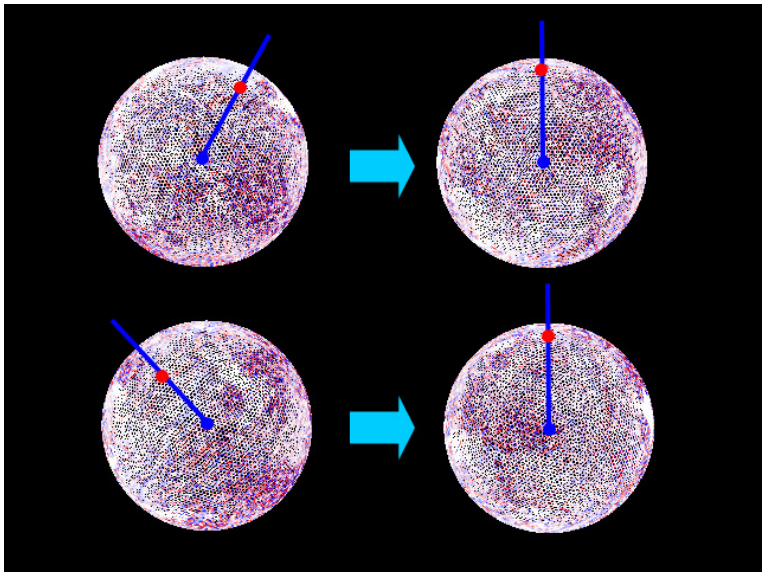


Figure 4. 1: Two SAIs of which the axes overlap each other

4.1.1 Calculation of the difference between two SAIs

Using two SAIs of two 3-dimensional shape models, we calculate the difference between two SAIs. We obtain the latitude and longitude at a node on one SAI surface. On another SAI surface, we define the node which is nearest by the latitude and longitude as the nearest neighbor node (See Fig. 4.2). Next, we calculate the absolute value of the difference between the simplex angle of the node on one SAI surface and the simplex angle of the nearest neighbor node on another SAI surface. And, we weight the absolute value by the function $\mathrm{Weight}(\theta)$ which is defined the latitude θ of the node on one SAI sphere surface when we assume the characteristic node as the North Pole.

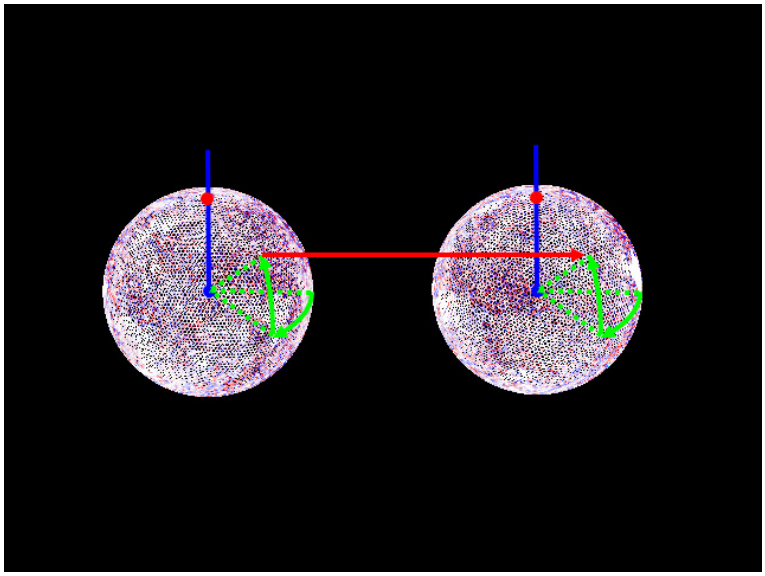


Figure 4. 2: A node on one SAI sphere surface and the nearest neighbor node on another SAI sphere surface

This function $\mathrm{Weight}(\theta)$ strengthens the absolute value between a node and its nearest neighbor node which are around the characteristic node. And, the value is $0 \le \mathrm{Weight}(\theta) \le 1$. We assume a characteristic node on one SAI surface as the North Pole (See Fig. 4.3). We define the absolute value between the latitude of the North Pole and the latitude of a node as $\theta°(0° \le \theta° \le 180°)$. Then, this function $\mathrm{Weight}(\theta)$ is the function decided by θ. When θ is small, in other words, when a node is near by the characteristic node, the value of the function $\mathrm{Weight}(\theta)$ is almost 1. When θ is large, in other words, when a node is far from the characteristic node, the value of the function $\mathrm{Weight}(\theta)$ is almost 0. However, if we don't weight, we define the function $\mathrm{Weight}(\theta)$ as constant value $\mathrm{Weight}(\theta) = 1$.

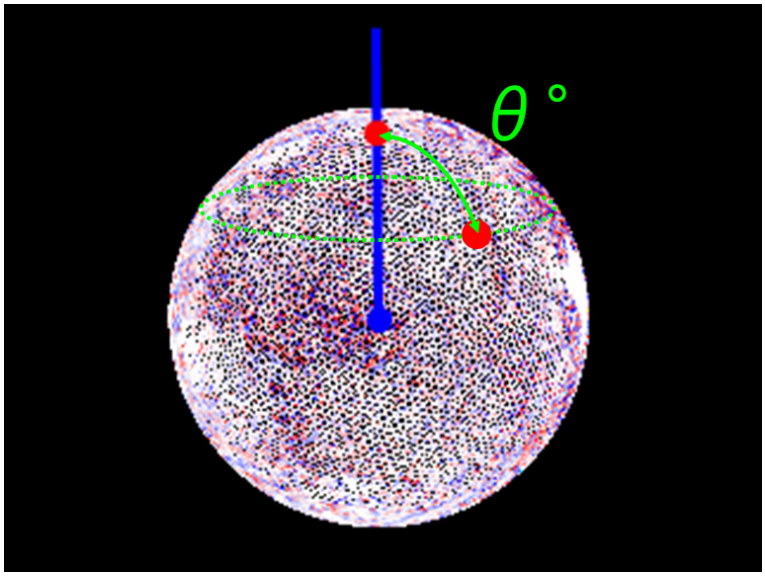


Figure 4. 3: Definition of θ

About the function $\mathrm{Weight}(\theta)$, we consider various variations, which is linear or non-linear, and continuity or discontinuity. Here, taking advantage of 'Lambert's cosine law'[8], we define the function $\mathrm{Weight}(\theta)$ as Equation (4.1). And, the graph of the function $\mathrm{Weight}(\theta)$ is as Figure 4.4.

$$\mathrm{Weight}(\theta)=\frac{1}{2}\cos(\theta)+\frac{1}{2} \quad (0^\circ \le \theta \le 180^\circ) \tag{4.1}$$

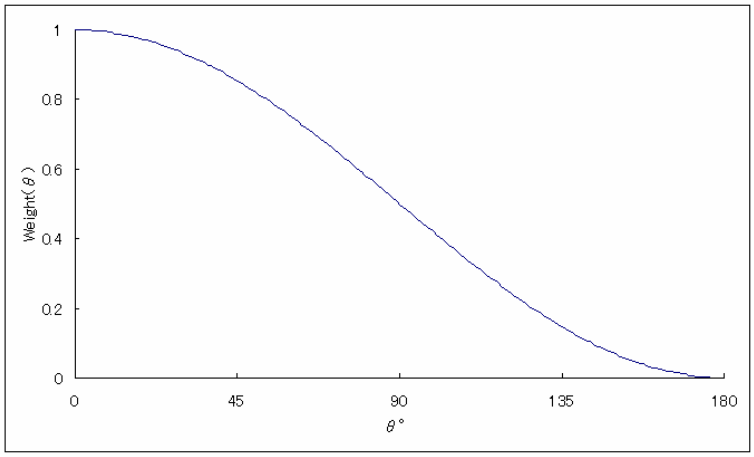


Figure 4. 4: Graph of the function $\mathrm{Weight}(\theta)$

Finally, about each nodes on one SAI, we calculate the weighted absolute value of the difference between two simplex angles. We define the sum of the weighted absolute

values as the difference between two SAIs.

$$Error = \sum_{\forall Node} \mathrm{Weight}(\theta) \times \left| SimplexAngle(Node) - SimplexAngle'(Node') \right| \tag{4.2}$$

4.1.2 Calculation of the distance between two 3-dimensional shape models

We calculate the distance between two 3-dimensional shape models, using the difference between their SAIs. About each SAI of two 3-dimensional shape models, two SAIs rotate around the center of SAIs to overlap each characteristic node on SAI sphere surface each other. Next, we define the line from a characteristic node on SAI surface to the center of SAI as the axis. One SAI rotates around the axis little by little, and we calculate the difference between one SAI and another SAI. While turning one revolution, we continue calculating the difference. Finally, when the difference is minimum at a rotation angle, we define the minimum difference as the distance between two 3-dimensional shape models.

4.2 Hierarchical cluster analysis

We perform hierarchical cluster analysis to analyze 3-dimensional shape models using the distances between each pair of 3-dimensional shape models.

In this section, they describe what hierarchical cluster analysis is and details of each method of hierarchical cluster analysis.

4.2.1 What hierarchical cluster analysis is

It is assumed that there are n objects $O_1, O_2, \cdots, O_n$. The similarity is $d_{ij}(1,2,\cdots,n)$ between O_i and O_j. d_{ij} is symmetrical ($d_{ij} = d_{ji}$).

The smaller the value is, the more similar O_i and O_j are. In this case, the index is called dissimilarity. In this section, they assume d_{ij} as dissimilarity.

Hierarchical cluster analysis is the method of analysis to generate the dendrogram from each dissimilarity d_{ij} between a pair of objects. The dendrogram is cut off at a section, and they can obtain n clusters. The small cluster near by each leaf of dendrogram is included in the large cluster near by the root of dendrogram.

Each step of hierarchical cluster analysis is as follows.

(1) There are n clusters included one object.

(2) They refer to the dissimilarity d_{ij}, and generate one cluster united two clusters which are the most similar.

(3) If there is only one cluster, then hierarchical cluster analysis is finished. Else, they go next step (4).

(4) They calculate the dissimilarity d_{ij} between new cluster generated at (2) and other clusters. And, they go back to (2).

Here, they can calculate dissimilarity d_{ij} without reference to the original data that they calculate d_{ij} from. This is called combinatorial method which is easy to calculate dissimilarity and is used generally.

4.2.2 Various methods of hierarchical cluster analysis

There are various calculation of dissimilarity which is calculated at step (4). Here, they describe each variation of hierarchical cluster analysis.

(1) Nearest neighbor method

They unite cluster p and cluster q, and they generate new cluster t. The number of objects organized each cluster is $n_p, n_q, n_t \left(= n_p + n_q\right)$.

They express the dissimilarity d_{tr} between cluster t and other cluster r using dissimilarity d_{pr}, d_{qr} between cluster p, q, and r. They define this dissimilarity as follows.

$$d_{tr} = \min\left(d_{pr}, d_{qr}\right) \tag{4.3}$$

When they apply this definition, dissimilarity between two clusters is defined by dissimilarity between two objects which are included each cluster and are most similar than all the other pairs of objects. Then, this method is called nearest neighbor method or single linkage method.

(2) Furthest neighbor method

When they unite cluster p and cluster q, and generate new cluster t. They define

dissimilarity d_{tr} between cluster t and other cluster r as follows.

$$d_{tr} = \max\left(d_{pr}, d_{qr}\right) \tag{4.4}$$

In contrast to the nearest neighbor method, they define dissimilarity between two clusters as dissimilarity between two objects which are included each cluster and are most dissimilar than all the other pairs of objects. Then, this method is called furthest neighbor method or complete linkage method.

(3) Group average method

In nearest neighbor method and furthest neighbor method, they define dissimilarity of new cluster united with a pair of clusters. But, these method are extreme, because the dissimilarity is calculated using maximum or minimum dissimilarity of all pair of objects included each cluster. Then, they define dissimilarity using the average of dissimilarity included cluster t and cluster r.

They express the relationship between before and after fusion as follows.

$$d_{tr} = \frac{n_p d_{pr} + n_q d_{qr}}{n_p + n_q} \tag{4.5}$$

(4) Centroid method

In centroid method, the dissimilarity between a pair of clusters is defined with the dissimilarity among the center of cluster. Here, let each object be m-dimensional observation value $\left(x_1, \cdots, x_m\right)$.

The center of gravity of cluster p, q is $\left(\bar{x}_1^{(p)}, \cdots, \bar{x}_m^{(p)}\right)$, $\left(\bar{x}_1^{(q)}, \cdots, \bar{x}_m^{(q)}\right)$. The center of gravity t of cluster $\left(\bar{x}_1^{(t)}, \cdots, \bar{x}_m^{(t)}\right)$ is expressed as follows.

$$\bar{x}_j^{(t)} = \frac{n_p \bar{x}_j^{(p)} + n_q \bar{x}_j^{(q)}}{n_p + n_q} \tag{4.6}$$

Moreover, squared Euclidean distance $E_{tr}^{\ 2}$ between the centers of gravity of cluster t and r is expressed as follows.

$$
\begin{aligned}
{E_{tr}}^2 &= \sum_{j=1}^{m}\left(\frac{n_p \bar{x}_j^{(p)} + n_q \bar{x}_j^{(q)}}{n_p + n_q} - \bar{x}_j^{(r)}\right)^2 \\
&= \frac{n_p}{n_p + n_q}{E_{pr}}^2 + \frac{n_q}{n_p + n_q}{E_{qr}}^2 - \frac{n_p n_q}{\left(n_p + n_q\right)^2}{E_{pq}}^2
\end{aligned}
\tag{4.7}
$$

They define $d_{ij} = {E_{ij}}^2$ as dissimilarity. Then, they renew dissimilarity as follow equation.

$$
d_{tr} = \frac{n_p}{n_p + n_q} d_{pr} + \frac{n_q}{n_p + n_q} d_{qr} - \frac{n_p n_q}{\left(n_p + n_q\right)^2} d_{pq} \tag{4.8}
$$

As described above, This method is meaning only when the dissimilarity is calculated based on squared-Euclidean-distance manner.

(5) Median method

Median method is the method that centroid method is simplified. In centroid method, they weight dissimilarities using the size of cluster n_p, n_q. However, at median method, they constantly weight not using the size of cluster. The equation is expressed as follow equation.

$$
d_{tr} = \frac{1}{2} d_{pr} + \frac{1}{2} d_{qr} - \frac{1}{4} d_{pq} \tag{4.9}
$$

These corresponds for the representative point of unified cluster to be measured dissimilarity between a pair of clusters by squared Euclidean distance between representative points.

(6) Ward method

They consider ith objects included cluster p as observation value ${x_{ji}}^{(p)}$ about the variable in Ward method [14]. They express sum of squares in cluster p as next equation.

$$
S_p = \sum_{j=1}^{m}\sum_{i=1}^{n_p}\left({x_{ji}}^{(p)} - \bar{x}_j^{(p)}\right)^2 \tag{4.10}
$$

They unite cluster p, q, and generate new cluster t.

Here, they define increase of sum of squares in cluster as ΔS_{pq}. This equation is as

follows.

$$\Delta S_{pq} = \frac{n_p n_q}{n_p + n_q} \sum_{j=1}^{m} \left(x_j^{(p)} - \bar{x}_j^{(p)} \right)^2 \tag{4.11}$$

In Ward method, it is desirable that sum of squares in cluster is smaller and smaller. At each step, they unite cluster p, q that increase of sum of squares in cluster ΔS_{pq} by uniting clusters is smallest. Because of this, they use ΔS_{pq} as dissimilarity d_{pq} between cluster p, q.

They unite cluster t which is united two cluster p and q, and other cluster r. The increase of sum of squares ΔS_{pq} is expressed as follows.

$$\begin{aligned} \Delta S_{tr} &= \frac{n_t n_r}{n_t + n_r} \sum_{j=1}^{m} \left(\bar{x}_j^{(t)} - \bar{x}_j^{(r)} \right)^2 \\ &= \frac{n_p + n_r}{n_t + n_r} \Delta S_{pr} + \frac{n_q + n_r}{n_t + n_r} \Delta S_{qr} - \frac{n_r}{n_t + n_r} \Delta S_{pq} \end{aligned} \tag{4.12}$$

The equation to renew dissimilarity is next equation.

$$d_{tr} = \frac{n_p + n_r}{n_t + n_r} d_{pr} + \frac{n_q + n_r}{n_t + n_r} d_{qr} - \frac{n_r}{n_t + n_r} d_{pq} \tag{4.13}$$

When each cluster includes one object, they use dissimilarity among the origin objects with $\frac{1}{2}$ of squared Euclidean distance.

(7) Flexible method

In each method described above, they renew dissimilarity. Using parameter α_p, α_q, β, γ, they can express a common equation as follows.

$$d_{tr} = \alpha_p d_{pr} + \alpha_q d_{qr} + \beta d_{pq} + \gamma \left| d_{pr} - d_{qr} \right| \tag{4.14}$$

Lance and Williams (1967) [9] advocated the method using above equation. They called a set of methods based on this equation as combinatorial method. Moreover, they proposed flexible method that meets the demand of the condition of the range which is $\alpha_p + \alpha_q + \beta = 1 : \alpha_p = \alpha_q : \beta < 1 : \gamma = 0$ about $\alpha_p, \alpha_q, \beta, \gamma$. Considering the above

condition, parameter β is only free. They use $-\frac{1}{4} \le \beta \le 0$, especially use $\beta = -\frac{1}{4}$.

(8) summary of hierarchical cluster analysis

About each method, they show the parameter at combinatorial method and feature at Table 4.1. Here, monotonicity of the dendrogram is shown monotonic increasing of dissimilarity when they hierarchically unite clusters. In this case, the branch of the dendrogram doesn't grow up reversely. And, space-contracting/dilating/conserving is the case of being near by other cluster/the case of being far from other cluster/the case of being constant from other cluster, when they unite two clusters and generate one new cluster. For example, in nearest neighbor method, space-contracting is happened. And, more uniting clusters, nearer by the distance from other clusters. Then, it is known that it is easy to generate the long chain cluster.

Table 4. 1: The correspondence the parameter at combinatorial method and each hierarchical cluster analysis method.

method	α_p	α_q	β	γ	monotonicity of the dendrogram	space-contracting, dilating or conserving
nearest neighbor	$\frac{1}{2}$	$\frac{1}{2}$	0	$-\frac{1}{2}$	○	space-contracting
furthest neighbor	$\frac{1}{2}$	$\frac{1}{2}$	0	$\frac{1}{2}$	○	space-dilating
group average	$\frac{n_p}{n_t}$	$\frac{n_q}{n_t}$	0	0	○	space-conserving
centroid	$\frac{n_p}{n_t}$	$\frac{n_q}{n_t}$	$-\frac{n_p n_q}{n_t^2}$	0	×	space-conserving
median	$\frac{1}{2}$	$\frac{1}{2}$	$-\frac{1}{4}$	0	×	space-conserving
Ward	$\frac{n_p+n_r}{n_t+n_r}$	$\frac{n_q+n_r}{n_t+n_r}$	$-\frac{n_r}{n_t+n_r}$	0	○	space-dilating
flexible	$\frac{1-\beta}{2}$	$\frac{1-\beta}{2}$	$\beta<1$	0	○	$\beta>0$ space - contractin g $\beta=0$ space - conserving $\beta<0$ space - dilating

4.3 Fast calculation of the distance between two 3-dimensional shape models

When we calculate the distance between two 3-dimensional shape models, we calculate the difference between their SAIs. There are n nodes including one SAI. The order of the calculation of the difference is $O(n^2)$ as described in Section 4.3.1. However, when n is large, we spend a long time of calculation. Then, we will solve this problem not using strict solution but using approximation solution. Concretely, from a node on one SAI, we obtain the neighbor node on another SAI using approximation solution with reference table whose the order of calculation is $O(1)$.

In this section, first, we describe the problem about the cost of calculation of the difference between two SAIs. Next, we describe how to search nearest neighbor node using approximation solution with reference table. Finally, we verify the effect and the

accuracy of fast calculation by comparing approximation solution with strict solution.

4.3.1 The cost of calculation of the difference between two SAIs

At calculation of the distance between two 3-dimensional shape models, we consider the order of calculation of the difference between their SAIs. The order of calculation of nearest neighbor node search by strict solution is $O(n)$, when there are n nodes. And, the order of calculation of the absolute value of the difference between two simplex angles is $O(n)$. Then, the order of calculation of the difference between two SAIs is $O(n^2)$. Here, the number of nodes included in an SAI using the experiment in this thesis is 20480. Then, we spend very long time for the calculation. However, taking advantage of reference table, we search nearest neighbor node in $O(1)$. Moreover, we calculate the difference between two SAIs in $O(n)$. This is fast calculation.

4.3.2 Reference table

We obtain simplex angle of nearest neighbor node by approximation solution using reference table. Beforehand, we construct reference table by mapping simplex angle of another SAI on a plane. About how to map simplex angle, we take advantage of 2-dimensional expression like a world map. Up to this time, there are a lot of variations of 2-dimensional expression of a world map [10][11][12]. In this thesis, we use the "Mercator projection", in 1569, because this is one of the simplest method and assumes the earth not ellipsoid by rotation but sphere.

The reference table is expressed on a 2-dimensional plane. One axis corresponds to latitude, and the other axis corresponds to the longitude (See Fig. 4.5). It is possible to easily access the target node using the reference table, if one knows its position in the latitude-longitude coordinate.

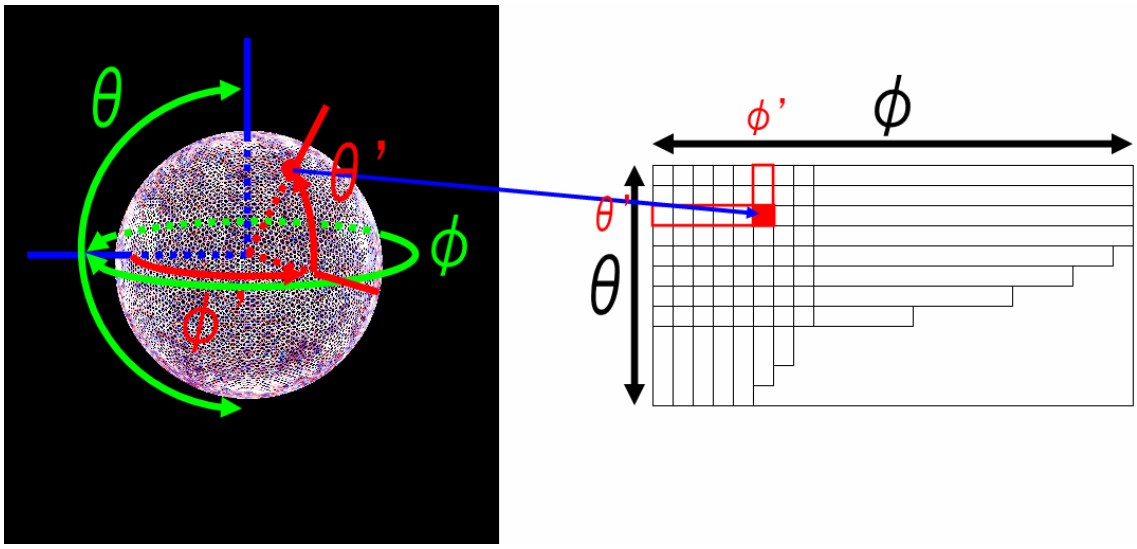


Figure 4. 5: Simplex angle reference table

Next, we describe how to generate reference table. We prepare a initial table whose cells are empty. About a cell for which we would like to substitute the value of simplex angle, we obtain latitude and longitude on SAI sphere surface. And, we obtain the node which is nearest by the cell. Then, we substitute the value of simplex angle of the node for the cell. And so on, we substitute simplex angles for other cells, and obtain reference table.

Other approximation solution of nearest neighbor node search is using k-d tree. There are n nodes in SAI. And, the order of calculation by k-d tree is $O(\log n)$. Because we expect the faster solution, we select approximation solution of reference table which is in the order $O(1)$.

4.3.3 Verification of accuracy and the effect of reducing the computation time

In this experiment, first, we examine the difference between two SAIs by strict solution and by approximation solution. And, we calculate the error between the differences by strict solution and by approximation solution, and verify accuracy. Second, we examine the effect of reducing the computation time by approximation solution compared with strict solution. And, we measure runtime of the calculation of the difference between two SAIs by both strict solution and approximation solution.

For this purpose, first, we prepare a several SAIs constructed by n nodes. Next, we select two SAIs from all SAIs at random. And, we calculate the difference between two SAIs by strict solution and approximation solution using reference table. Moreover, we measure the error occurring from the calculation by approximation solution. And, we measure each runtime by strict solution or approximation solution. The result of this experiment is Table 4.2.

Table 4. 2: Error and the rate of reducing runtime by approximation solution compared with strict solution

a pair of 2 SAI Images	the difference between 2 SAI Images			runtime		
	strict solution d_{str}	approximation solution d_{aprx}	error $\frac{\lvert d_{aprx} - d_{str} \rvert}{d_{str}}(\%)$	strict solution $t_{str}(\mathrm{sec})$	approximation solution $t_{aprx}(\mathrm{sec})$	ratio $\frac{t_{aprx}}{t_{str}}(\%)$
(1)	13,349.9	13,389.0	0.293	14,555	2	0.014
(2)	13,831.9	13,844.7	0.093	15,165	1	0.007
(3)	14,764.6	14,769.3	0.032	17,174	2	0.012
(4)	14,511.8	14,522.2	0.072	16,746	2	0.012
(5)	13,832.6	13,853.9	0.154	15,187	2	0.013

The error of approximation solution compared with strict solution is less than 0.3%. We regard approximation solution using reference table as accuracy enough. Also, runtime of approximation solution compared with strict solution is reduced about 0.01%. Because strict solution is $O(n)$ and approximation solution is $O(1)$, the rate of reducing runtime is about $1/n = 1/20480 \cong 0.005(\%)$ in theory. In practice, the effect of reducing runtime is as half as the effect of reducing computation time in theory, because the calculation time measured by performing actual calculation justifies the consideration. But, we confirm the effect of reducing computation time by approximation solution using reference table.

About approximation solution using reference table, before calculation of the difference between two SAIs, we need to generate reference table beforehand. Here, because the number of nodes in SAI is $n = 20480$, we generate reference table that latitude and longitude is $360 \times 180 = 64800$. It spends about four seconds to generate one reference table. Moreover, once we generate a reference table of an SAI, we can use the reference table to calculate the difference from other SAIs. We can use the reference table. That

is we, need only to generate the reference table once.

4.4 Experiment

In this section, we estimate phylogenetic relationships of fowls by analysis using SAIs. We prepare a number of fowl skull datum as 3-dimensional shape models. Next, we generate SAIs from these 3-dimensional shape models using our proposed method in this thesis. Using the SAIs, we calculate the distances between all pair of 3-dimensional shape models. We regard these distances as dissimilarity, perform hierarchical cluster analysis, and generate the dendrogram of relationships of fowl skulls. Finally, from the generated dendrogram, we estimate phylogenetic relationships of fowls.

In this section, first, we describe how to obtain models of fowl skulls. Finally, we analyze using SAIs and illustrate the result of this analysis.

4.4.1 Researches about estimation of phylogenetic relationships of fowls

Japan is inhabited by the unique native fowls called Nihon-Kei inhabits. Among these, 17 breeds, such as the Syamo (Syamo means a fighting fowl in Japanese.) and Gifu-Jidori (Jidori means a domestic fowl in Japanese.), have been designated protected breeds in Japan. Although these fowls originated in Southeast Asia and China, they were imported to our country during the Yayoi period, which took place hundreds of years B.C. The breeds were subsequently improved for various purposes of use, including satisfying the Japanese fondness for meat and cockfights. As a result, many characteristic breeds were produced.

Hereditarily, these Japanese native fowls are important breeds. Recognizing their importance made some amateur lovers preserve these fowls. Later on, the necessity for their preservation by public means has been recognized. Therefore, various investigations and research on their origin and on their phylogenetic relationships have been performed.

Recently, analysis of relationships by blood according to ancient documents along with analysis of forms, blood types, or DNA have been performed by Oana et al. [15],

Hayashi et al. [16], Nishida et al. [17], Okada et al. [18], Hashiguchi et al. [19], Tanabe et al. [20], and Takahashi [21]. The research can be classified into three categories: (1) analyzing relationships by blood directly according to ancient documents etc., (2) analyzing fowls' genes using their blood, e.g., DNA, etc., and (3) analyzing fowls' shapes using the morphological analysis methods.

Among morphological analysis methods, i.e., analyzing interspecies differences in their body forms, osteometric methods are common. These methods analyze the difference in a shape by choosing characteristic parts among skeletal specimens. Concretely speaking, they measure distances for the specifics points precisely and compare the corresponding distances in different breeds by the multivariate analysis such as the principal component analysis (PCA).

In these methods, only distances of the specific points of the specimens can be obtained, but it is difficult to analyze form of the specimens totally and to distinguish individual differences such as the size of each individual, and the interspecies morphological differences. Then, we applied our proposed method for the morphological analysis.

4.4.2 Generating of 3-dimensional shape models

In this section, we describe how we measured the real objects such as the skeletal specimens of fowls' skulls with a laser scanner and generated their precise 3-dimensional shape models.

First, we describe the technique of obtaining a 3-dimensional picture, called a range image, from which depth information was mapped onto each pixel. Next, we describe the method of alignment of the range images from different positions into a single coordinate system. Finally, we describe the method for generating a single 3-dimensional model by merging range images which have already been aligned.

(1) Measurement

A laser scanner is a device for obtaining a range image by irradiating a real object with a laser beam. This measurement gives us the accurate length from viewpoint to the surface of the real object. The lengths are mapped onto each pixel of the range image as

depth information. In this thesis, we used a laser scanner called VIVID910, a product of KONICA MINOLTA, Inc.

Since the position of a laser scanner must be fixed when measuring a real object, the shape of the object's back and the portion that is in shade cannot be acquired in one measurement. Thus, it is necessary either to rotate the object or to change scanner's position for measuring from sufficient multiple positions to enable us to measure the entire object surfaces.

In this thesis, we measured the specimens of skulls of fowls. These samples included the following breeds of native Japanese fowl: five Gifu-Jidoris, five Satsuma-Doris (Dori or Tori means a fowl in Japanese.), five Syamos, five Ko-Syamos (Ko- means small in Japanese.), five Hinaidoris, five Shokokus, two Koeyoshis, five Tomarus, and five Totenkos. Also included were four European White Leghorns, five Red Jungle-fowls, one Grey Jungle-fowl, and one Ceylon Jungle-fowl. The breeds are the most standard and popular egg-breeds and we used them as benchmarks in the analysis of the phylogenetic relations.

We measured the skulls of 53 individual fowls belonging to 13 different species. Considering complexity of a shape of each individual fowl, measurements from 11 different viewpoints are enough to model the fowl.

(2) Alignment

The range images described in the previous section cannot be merged into a single 3-dimensional shape model because they are placed in different camera coordinate systems that depend on the position of the laser scanner at the time of measurement.

Until now, the method of alignment of such range images into a single world coordinate system involved the use of a software operation proposed by Besl and McKay [22] and Nishino et al. [23]. In our research, using this method, we aligned 11 range images per each individual into a single world coordinate system.

(3) Merging

The range images aligned into a single coordinate system in the previous section are a

simple set of coordinates of 3-dimensional points without connectivity, face information, etc. In those range images, since the points that were measured from different viewpoints were measured repeatedly, the point density changes with places. Therefore, it is difficult to analyze the geometrical shape of the range images.

To solve this problem, we considered merging range images already aligned and representing the geometrical shape of the real object as a mesh model with face information. Until now, Wheeler et al. [24] proposed a method of merging range images and generating a mesh model using the Signed Distance Field. In our research, by using this method, we generated the mesh model of each individual.

Figure 4.6 shows an example of merging. This is a merged mesh model of a Gifu-Jidori. Such data obtained by merging are satisfied with all requests for "3-dimensional shape model," i.e., can be applied to our proposed method.

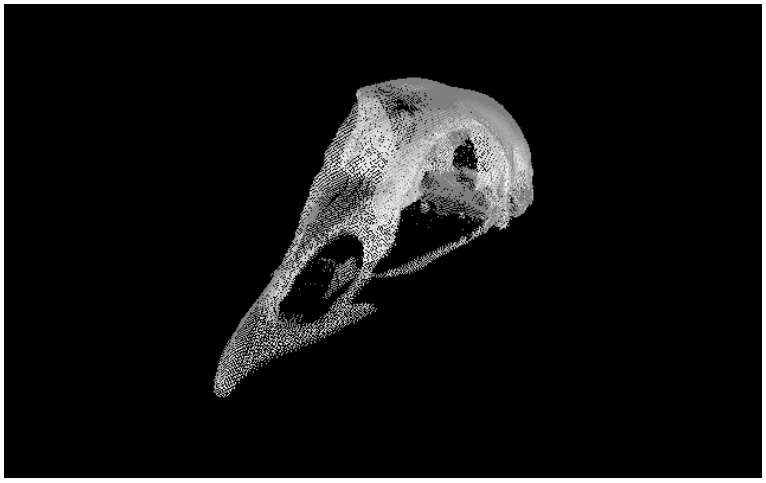


Figure 4. 6: Merged mesh model of a Gifu-Jidori

4.4.3 Analysis using SAIs and the result

We generated SAIs from 3-dimensional shape models of fowl skull datum. And, we calculated the distances between all pair of 3-dimensional shape models. Regarding the distance as dissimilarity, we generated the dendrogram of relationships of fowl skulls by performing hierarchical cluster analysis.

We prepared 53 samples (13 species) of fowl skull datum. We generated SAIs with

$20 \times 4^5 = 20480$ nodes. Then, when we calculate the distance between 3-dimensional shape models, we need a characteristic node every SAI. Here, we used the specific region named "stop." The "stop" is regarded as an important region for classifying the skull shape of not only fowls but also of many animal species. We performed flexible method ($\beta = -1/4$) of hierarchical cluster analysis. Then, we generated the dendrogram of relationships of the breeds (See Fig. 4.7). The horizontal axis is expressed the distance between clusters.

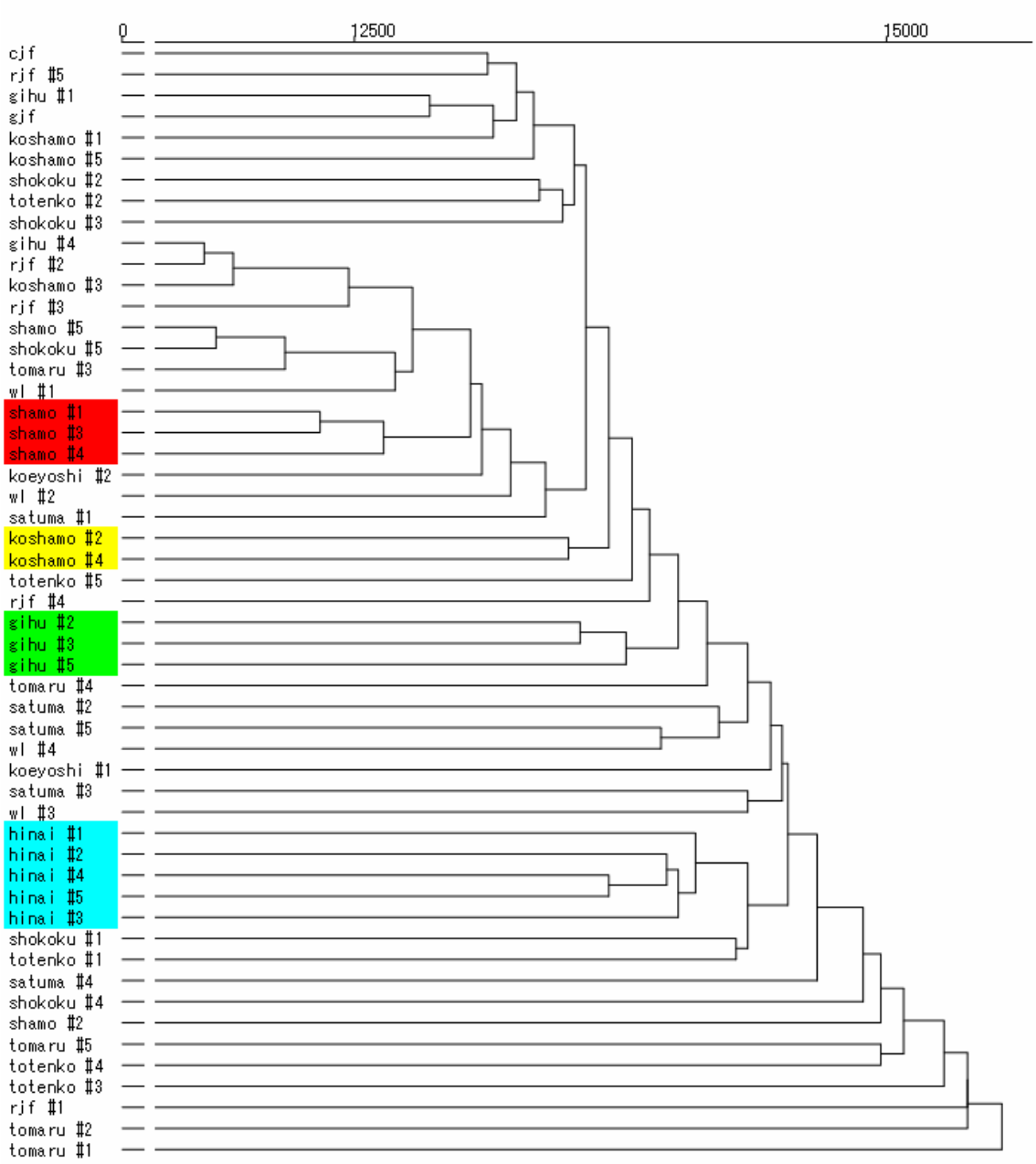


Figure 4. 7: The dendrogram of the relationships of the breeds

4.4.4 Discussion

From the dendrogram (See Fig. 4.7), we can see generated clusters of {shamo #1, shamo #3, shamo #4}, {koshamo #2, koshamo #4}, {gihu #2, gihu #3, gihu #5}, {hinai #1, hinai #2, hinai #4, hinai #5, hinai #3}.

Other samples are not classified into one cluster, if they belong the same species. We consider that this unclassified is occurred in the cause of partial lack of fowl skull datum. Figure 4.8 shows one example of the complete specimens of the fowl skull. However some specimens, as shown in Fig. 4.9, lack some portions (in this case, the portion of the jaw lacks). Considering the case to generate the SAIs from the complete specimen and the specimen which is obtained by removing some portion from the same specimen. Two SAIs obtained from them should be quite different in spite that they are from the same origin.

However, partial SAIs far from the area corresponding to the lacking portion should be similar to each other. That is why we use the weighted distance between two SAIs as the dissimilarity of the 3-dimensional shape models. Fortunately, the "stop" area of fowls skills hardly ever lack.

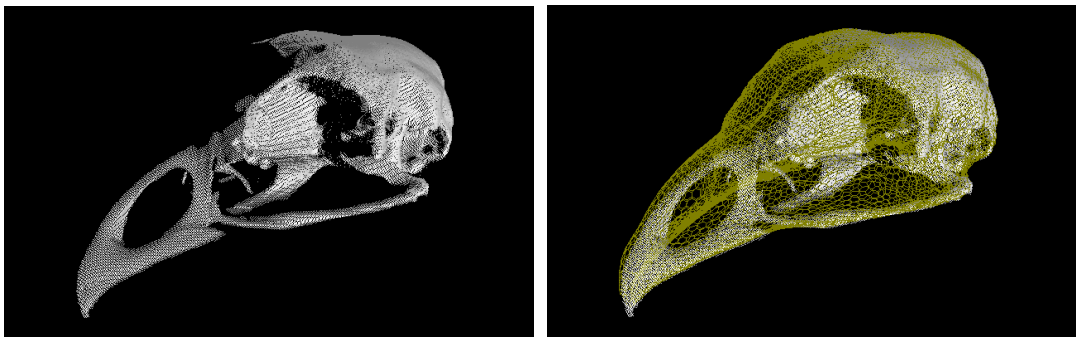


Figure 4. 8: Fowl skull with no lack and convergence circumstances of the mesh

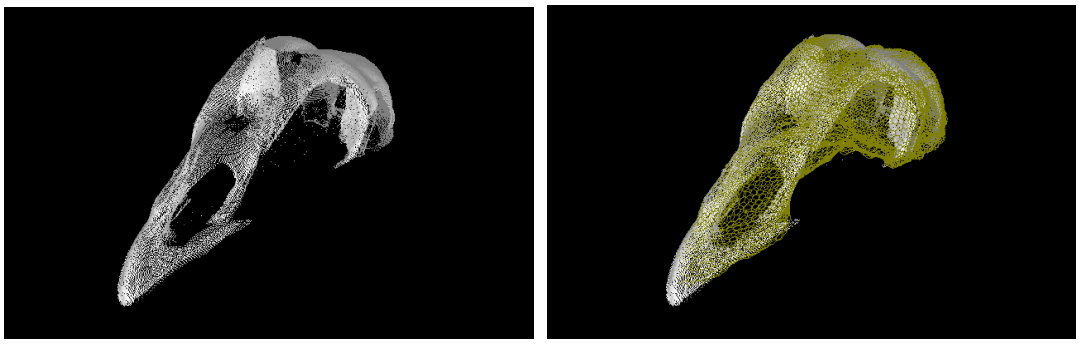


Figure 4. 9: Fowl skull with partial lack and convergence circumstances of the mesh

Chapter 5 Conclusion and Future Work

In this thesis, we improved the spherical-attribute-image (SAI) method for complicated objects with depressions and partial fractures. For achieving that, we proposed two novel methods: one is to take advantage of a data set called “guiding intermediate shapes” in order to robustly obtain the semi-regular mesh which sufficiently approximates the original 3-dimensional shape model with depressions. The other is to define the distance between two 3-dimensional shape models using appropriately weighted differences between all pairs of nodes on their SAIs in order to reduce the influence of partial fractures. In actuality, we verified the effectiveness of our proposed methods.

First, we described the concept and the algorithm of the original SAI method. When SAIs are generated from 3-dimensional shape models, it is necessary to obtain the semi-regular mesh which has the two features: the approximation of 3-dimensional shape model's surface and local regularity. However, it is sometimes impossible for this method to obtain such a mesh. We guessed that this impossibility is caused by realizing the two quite conflicting features simultaneously, not the two-step manner. Then, we divided this obtaining process into two steps. Concretely we first obtain a series of guiding intermediate shapes which gradually deform from the initial spherical mesh model to the target model; in this time we do not consider the local-regularity at all. Then we fit the initial mesh model onto a series of models in order considering the local regularity. As a result, we made it possible to obtain the mesh satisfied with both features.

Next, we compared our proposed method for generating SAIs with the original SAI method with respect to their accuracy and computational time using a virtual 3D model. We actually illustrated the effectiveness of our proposed method using some statistical measures, e.g. variation. Moreover, our method could use coarse point cloud as the guiding intermediate shapes, since the shapes only need to roughly guide the deformed mesh. As the result, we confirmed that our method achieved high-speed generation of SAIs through the comparison.

Next, we redefined the distance between two 3-dimensional shape models using their SAIs. We appropriately weighted the difference between two simplex angles, which are geometrical attributes mapped on the SAIs, in accordance with their positions on the SAIs in order to reduce the influence of the partial fractures. And we proposed a fast calculation of the distance using "Reference Table;" the order of the calculation is O(n), where n is the number of nodes of SAIs.

After that, we performed the osteometrical analysis of fowls using our proposed methods and hierarchical cluster analysis; the samples consisted of the skulls of 53 fowls (13 kinds of species) and some of them lacked a jaw part. We actually illustrated the result using a dendrogram.

Considering the case of existing partial fractures again, one definitely reduces the influence of them at the analysis. It is generally considered that there are the two methods for realizing that as follows: (1) limitation of the areas which are used for the analysis and (2) use of distances which is a sum of appropriately weighted differences between all pairs of nodes on their SAIs, that is, our proposed method. Note that, the original SAI method may accept the former method. Although the question "which is better?" is still an open problem, our proposed method may have an advantage to cleverly, i.e., without no pre-processing, reduce the influence.

References

[1] Gauss, K.F., General Investigations of Curved Surfaces, Raven Press, New York, 1965.

[2] Horn, B.K.P., “Extended Gaussian Image”, Proc. Of IEEE, Vol. 72, No. 12, pp.1671-1686, December 1984.

[3] M. Vasilescu and D. Terzopoulos. Adaptive Meshes and Shells. Proc IEEE CVPR. pp. 829-832, 1992.

[4] L. Cohen and I. Cohen. Finite Element Methods for Active Contour Models and Balloons from 2D to 3D. IEEE Trans. PAMI. Vol. 15, No. 11, pp. 1311-1147, November 1993.

[5] Y. Chen and G. Medioni. Surface Description of Complex Objects from Multiple Range Images. Proc. IEEE CVPR. pp. 153-158, June, 1994.

[6] T. McInerney and D. Terzopoulos. Deformable Models in Medical Image Analysis: A Survey. 1996. Medical Image Analysis 96/1.

[7] K. Ikeuchi and M. Hebert, “Spherical Representations: from EGI to SAI”, CMU-CS-95-197, October 1995.

[8] Johann Heinrich Lambert, “Photometria”, 1760.

[9] Lance, G. N. and Williams, W. T. (1967). A general theory of classificatory sorting strategies, I. Hierarchical systems, The Computer Journal, 9, 373-380.

[10] Snyder J. P., 1993, Flattening the Earth - Two thousand years of map projections, The University of Chicago Press

[11] Snyder J. P. and Voxland, P. M., 1989, An Album of Map Projections, U.S. Geological Survey Professional Paper 1453.

[12] Canters, Frank, and Hugo Decleir, 1989, The World in Perspective: A Directory of World Map Projections, Chichester, Eng.: John Wiley and Sons.

[13] Review, 38, 406-427 Tryon, R. C. 1939)Cluster Analysis'New York: McGraw-Hill. Tufte, E. R.

[14] J. Ward. Hierarchical grouping to optimize an objective function. Journal of the American Statistical Association, 58:236-244, 1963.

[15] Oana H. (1951) History of Japanese domestic fowl, Nihon-kei Kenkyusha

[16] Hayashi Y., Nishida t., Fujioka T., Tsugiyama I., Mochizuki K. and Tomimoto M. (1982) Measurement of the Skull of Jungle and Domestic Fowls. Jpn. J. Vet. Sci. 44(6): 1003-1006

[17] Nishida T., Hayashi Y., Fujioka T,. Tsugiyama I. And Mochizuki K. (1985) Osteometrical Studies on the Phylogenetic of Japanese Native Fowls, Jpn. J. Vet. Sci. 47(1):25-37

[18] Okada I., Yamamoto Y., Hashiguchi T., Ito S. (1984) Phylogenetic studies on the Japanese native breeds of chickens, Japanese Poultry Science, 21:318-329

[19] Hachiguchi T., Tsuneyoshi M., Nishida T., Higashiuwatoko H., Hiraoka T. (1981) Phylogenetic relationships determined by the blood protein types of fowls. Jpn. J. Zootech. Sci. 52(10):713-729

[20] Tanabe Y., Iida T., Yoshino H., Shinjo A., Muramatsu S. (1991) Studies on the phylogenetic relationships of the Japanese native fowl breeds 5. The comparisons among native fowl breeds in Japan and its adjacent areas and European and American fowl breeds. Jpn. Poult. Sci., 28:266-277

[21] Takahashi H. (1997) Genetic relationships between Japanese chicken breeds on micro satellite DNA polymorphisms. J. Anim. Genet., 25(2):53-59

[22] Paul J. Besl, Member, IEEE, and Neil D. McKay, "A Method for Registration of 3-D Shapes", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 14, No. 2, February 1992.

[23] Taku Nishikawa, Ko Noshino, Yoichi Sato, and Katsushi Ikeuchi, "Constructing a 3D Model Using a High Resolution Range Sensor", Proceedings of the Virtual Reality Society of Japan Fourth Annual Conference, pp199-202, September, 1999.

[24] M. D. Wheeler, Y. Sato, K. Ikeuchi, "Consensus surfaces for modeling 3D objects from multiple range images", Proc. International Conference on Computer Vision, January 1998.